(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
8 February 2001 (08.02.2001)

PCT

(10) International Publication Number
WO 01/09875 A1

(51) International Patent Classification[7]: G10H 1/00

(21) International Application Number: PCT/US00/21019

(22) International Filing Date: 31 July 2000 (31.07.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/365,475 2 August 1999 (02.08.1999) US

(71) Applicant *(for all designated States except US)*: **DYNAMIX DIRECT, INC.** [US/US]; Suite 101, 22736 Vanowen Street, West Hills, CA 91507 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **DEKEL, Boaz, Desau** [IL/US]; 23515 A Hatteras Street, Woddland Hills, CA 91364 (US). **SHUB, Ronnie** [IL/IL]; Haruzim 71 A, 60917 Israel (IL). **KENNETH, Ronnie** [IL/IL]; Haruzim 71 A, 60917 Israel (IL). **AVERBUCH, Apolinari, Nir** [IL/US]; 21315 Lopez Street, Woodland Hills, CA 91364 (US). **ZVULUNY, Averbuch** [IL/IL]; Shlomzion Hamalka 40, 62266 Tel-Aviv (IL).

(74) Agent: **O'BANION, John, P.**; O'Banion & Ritchey LLP, Suite 1550, 400 Capitol Mall, Sacramento, CA 95814 (US).

(81) Designated States *(national)*: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW) , Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM) , European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE) , OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

WO 01/09875 A1

(54) Title: ONLINE COMPOSITION AND PLAYBACK OF AUDIO CONTENT

(57) Abstract: A method and system for the composition of audio content on the Internet where pre-recorded audio components, stored on a web application server (14), are assembled to compose audio and/or video content using various forms of information provided by the user (16, 18, 20). The web server (14) will use this information to create the composition and provide the user (16, 18, 20) with an opportunity to review and/or modify the greeting. Once the composition is complete, the user can designate a recipient of the composition. The web server (14) then sends an e-mail message to the designated recipient and invites them to visit the web site to listen to the composition where the composition can be retrieved.

TITLE OF THE INVENTION

ONLINE COMPOSITION AND PLAYBACK OF AUDIO CONTENT

CROSS-REFERENCE TO RELATED APPLICATIONS

Not Applicable

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

Not Applicable

REFERENCE TO A MICROFICHE APPENDIX

Not Applicable

NOTICE OF MATERIAL SUBJECT TO COPYRIGHT PROTECTION

All of the material in this patent document is subject to copyright protection under the copyright laws of the United States and of other countries. The owner of the copyright has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the United States Patent and Trademark Office file or records, but otherwise reserves all copyrights whatsoever.

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates generally to internet audio and more particularly to a system and method which provides internet users with the ability to compose, send, and play audio content such as greeting cards, gifts, computerized music compositions, and online music.

2. Description of the Background Art

There are several Internet-based audio technologies which facilitate the download and playback of audio content over the web. Typical sites with internet audio technology deliver pre-recorded fixed content, or live content controlled by the publisher. User control on these sites is generally limited to the addition, substitution, or removal of certain audio tracks. Users have been unable to create customized audio content on the Internet and play it back, or deliver that custom

content to other Internet users.

Therefore a need exists for an internet based solution that provides users with the ability to perform audio composition, production, transmission, and playback within an easy to use framework. The online composition and playback system in accordance with the present invention satisfies that need, as well as others, and overcomes deficiencies in previously known techniques.

BRIEF SUMMARY OF THE INVENTION

The present invention is a method and system for the composition of audio content on the Internet. The method uses pre-recorded audio components, stored on a web application server, which are assembled to compose audio content for playing through the use of standard Internet client software (browsers) containing software plug-ins.

By way of example, and not of limitation, the invention is hosted on a web server accessible to Internet users (customers) via a client computer. Using various forms of information provided by the user as well as user selections, a customized "greeting" or composition containing music, jokes, announcements and other audible and/or visual information is prepared by the web server. The user has access to various mixes, clips, jingles, songs, etc. spanning a wide variety of subjects for customization. While synthesized music could be provided, the music is preferably orchestrated by artists who play and/or sing various backgrounds and lyrics. The web server will use this information to create the greeting and provide the user with an opportunity to review and/or modify the greeting. Once the composition is complete, the user can designate one or more recipients for the composition. The web server then sends an e-mail message to the designated recipients and invites them to visit the web site to listen to the composition. When a designated recipient visits the web site, the recipient's computer retrieves (downloads) data which is unique to that recipient for playing the composition. The web server maintains a client information file for each designated recipient that determines what information will be retrieved by the recipient. Data is stored in terms of audio components and two additional descriptor files. The audio components consist of a primary component and one or more secondary components. The primary component contains background instrumentals and/or sound effects, mixed with vocals. The secondary

components contain vocals. The descriptor files contain sequencing and synchronization information. This data is downloaded to the recipient station as individual components and assembled at the recipient station. In addition, storage of playback audio content as separate components eliminates the need for the greeting to be stored on the web server as a distinct and separate composition. In other words, the descriptors specify which data components are individually accessed, as well as their sequence, to make up the complete composition.

Additionally it is contemplated that instead of requiring the designated recipient to return to the web site in order to playback the composition on multiple occasions, the greeting can be made available for download, or a CD created, with the composition and sent to the designated recipient.

An object of the invention is to enable internet users to compose custom audio components which may be sent over the internet to other internet users.

Another object of the invention is to provide interaction in real-time, such that Internet users can construct audio sequences dynamically and listen to them within seconds of their construction.

Another object of the invention is to provide for the construction of contiguous audio from multiple audio components.

Another object of the invention is to provide for dynamic audio sequence construction created from a set of pre-recorded audio components, such as high-quality studio-recorded audio sequences.

Another object of the invention is to allow internet users to create audio components maintained as separate channels which can be mixed as a time-based sequence, rather than as continuous channels.

Another object of the invention is to allow internet users to create audio segments that are used as parameters designed to fit into pre-defined audio sequence time slots (i.e. "...Happy birthday ____, Happy Birthday...").

Another object of the invention is to provide internet users with the ability to automate and synchronize the mixing of real-time audio content.

Another object of the invention is to allow internet users or recipients to perform client-side assembly of audio playback, wherein audio components are downloaded over the internet and the client controls the assembly of the components into a single audio sequence.

Another object of the invention is to sell downloadable versions of the composed audio content, as well as CDs containing a high-quality version of this content.

Further objects and advantages of the invention will be brought out in the following portions of the specification, wherein the detailed description is for the purpose of fully disclosing preferred embodiments of the invention without placing limitations thereon.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will be more fully understood by reference to the following drawings which are for illustrative purposes only:

FIG. 1 is a system level function block diagram of the invention where hardware elements are shown connected to the internet.

FIG. 2 is a functional block diagram of the architecture within the application server of the invention.

FIG. 3 is a functional block diagram of the recording process of the invention.

FIG. 4 is a functional block diagram of background track generation according to the invention.

FIG. 5 is a functional block diagram of the setup of the Sequencer/Editor during the processing of secondary audio components according to the invention.

FIG. 6 is a block diagram of the functions within a representative studio set-up for the recording of audio components according to the invention.

FIG. 7 describes the content of the synchronization descriptor and its timing relationship to the waveform diagram of the mixed background track (primary component) according to the invention.

FIG. 8 is a flow diagram of the composition process according to the invention.

FIG. 9 is a flow diagram of the playback process according to the invention.

FIG. 10 is a network connection diagram of a representative network according to the invention.

FIG. 11 is a logic diagram of selection rules for the audio component files according to the invention.

FIG. 12 is a representative sample browser composition screen with pull-down menus.

FIG. 13 is a representative diagram of the audio repository sequence tree according to the invention.

FIG. 14 is a representative diagram of the audio repository parameter tree according to the invention.

FIG. 15 is a diagram of a representative database schema for a portion of the system according to the invention.

FIG. 16 is a diagram of a representative database schema for another portion of the system according to the invention.

FIG. 17A is a flow diagram of a user traversing the Mixme.com site according to the invention in which Flash mode is available.

FIG. 17B is a flow diagram of a user traversing the Mixme.com site which continues from FIG. 17A.

FIG. 18A is a flow diagram of a user traversing the Mixme.com site according to the invention in which Flash mode is not available.

FIG. 18B is a flow diagram of a user traversing the Mixme.com site which continues from FIG. 18A.

FIG. 19 is a representative sample browser selection/composition screen with menus.

DETAILED DESCRIPTION OF THE INVENTION

Description of Preferred Embodiment

Referring more specifically to the drawings, for illustrative purposes the present invention is embodied in the apparatus and method generally shown in FIG. 1 through FIG. 19. It will be appreciated that the online composition and playback system of the invention may vary as to configuration and as to details of the parts, and that the method may vary as to the steps and their sequence without departing from the basic concepts as disclosed herein.

System Architecture

A system 10 for online composition and the playback of audio files according to the invention is shown in FIG. 1. The system comprises three main components:

(a) Recording application 12 - provides offline preparation and storage

of audio files for use over the web.

(b) Web server application 14 - provides online composition and transmission of audio files over the internet.

(c) Web client application 16, 18, 20 - provides online assembly and playback of audio files on the client's hardware. Although only three clients are shown for clarity, a vast number of client applications can be supported.

The recording application 12 includes a recording studio 22 in which audio is created and modified, along with an audio repository 24 wherein the created audio is stored as files for later retrieval. Web server application 14 accesses the stored audio files by means of the connection 26, which is any conventional data transfer link.

Audio files are served up over an IP connection 28 to the internet 30, through which the audio files subsequently pass over any of the local IP segments 32, 34, 36 to a respective client application 16, 18, 20 as determined by user interaction with the client application.

Audio Server Architecture

The audio server application uses standard web server application guidelines which may be implemented in a variety of server application environments such as Common Gateway Interface (CGI) and Active Server Page (ASP). FIG. 2 shows an example of an ASP-based implementation of the audio application server 50. The audio application server comprises two primary components, the audio index database host 52, which acts as a filing system for information about the audio components, and a web host 54 which operates from scripts to perform user directed audio processing. The database host contains an audio index database 56 containing indexed audio file information about stored audio components. A database server 60 within the database host accepts queries from the web host 54 though a data connection 62. The queries are executed against the data in the audio index database 56 to generate a result set which is returned to the web host 54 for further processing. The database server is shown as an SQL (Structured Query Language) type server. Although SQL is preferred within the embodiment due to its prevalence, flexibility, and speed, it must be noted that any form of database and query language can be used within the system to similar effect. The web host 54 contains an audio application server

68. The application server has a data connection 62 to the database server 60, a data connection 66 to an audio repository 64, and another data connection 70 to a scripting engine 72 (herein shown an Active-X Scripting engine). The audio application server takes audio processing commands from the scripting engine 72 and performs data storage, composition, and retrieval functions on the audio repository 64 which it manages. The audio repository 64 is a file system that contains a variety of audio sequences stored as files. It should be noted that the generally large and variably sized segments of audio data held within the repository need not be indexed (e.g. it can be organized as a flat file system). The SQL server database provides an indexing mechanism on the files in the audio repository. The scripting engine 72 performs various scripts from within the web pages hosted on the HTTP web server 76, and in accordance with those scripts directs the audio application server 68 to perform a variety of audio operations relative to the audio repository 64. The audio repository 64 is stored within a (non-indexed) standard hierarchical file directory structure. The audio index database 56 provides the indexing mechanism which thereby eliminates directory structure dependency that some indexing mechanisms are sensitive to.

Audio Client Architecture

The online audio client application (not shown) uses standard Internet browsers such as Netscape™ or Microsoft's Internet Explorer™, assisted by standard plug-in components, capable of audio playback, executing scripts, such as Java scripts, and downloading from the server. The online audio client application is hosted on a web client, such as the representative web clients 16, 18, 20, shown at the bottom of FIG. 1.

Functional Description

Audio Application Server

The audio application server performs audio functions which accept the desires of the user as they are entered via the web pages/scripting engine, and performs various operations on the audio repository as looked up via the database, to provide the desired audio content and functions desired by the user. Specifically the audio application server is responsible for the following:

(a) Serving audio selection/configuration pages.

(b) Composing an audio sequence by selecting the necessary audio

components.

(c) Transmitting all the necessary audio components, along with descriptor files and Java scripts that define how the audio components should be activated.

Audio Application Client

The audio application client interfaces with the user in relation to the web pages of the audio application. This interface collects user input and generates audio and video feedback to the user. The audio application client is responsible for the following:

(a) Selecting/configuring the audio sequence by allowing the user to interact with the application server.

(b) Sending a request for an audio sequence.

(c) Receiving an audio sequence and parsing it.

(d) Activating the appropriate plug-in components which are capable of playing back the audio components.

Process Flow

Recording

FIG. 3 is a high-level diagram of the process flow for the recording process 80. An audio background track is generated 82 with a Sequencer/Editor, which is bounced 84 so that the background track is combined (mixed) with the main (primary) audio component to produce a mixed background track. The resultant audio may be optionally wave converted 86, which can reduce the audio file size and/or used to improve sound quality in certain online audio players. Audio may then be encoded 88 in any desired encoding scheme, such as WAV, MP3, etc. The audio component is then stored within the audio repository 90 for later use. Synchronization descriptors are generated 94 to synchronize activities within a particular audio sequence. The synchronization descriptors are stored within the audio repository 90. Sequence descriptors are generated 96 to define the order in which audio components are embedded (mixed) within a background track. Sequence descriptors are also stored within the audio repository 90.

Generate Background Track

An audio background track is generated at 82 in FIG. 3 through the use of an Integrated MIDI & Digital Audio Sequencer/Editor 100, or the like, as shown in

FIG. 4. Referring also to FIG. 4, the input consists of one or more sub-tracks 104, 106, 108, which are encoded as a MIDI, or audio channel, and may correspond to a separate instrument. The sub-tracks 104, 106, 108 are mixed individually by a mixer 114, and combined into a background track, comprising left 110 and right 112 output channels.

Generate Audio Components

Audio components are generated at 92 in FIG. 3 such that they can be mixed online with the background track and/or inserted into pre-defined time slots within the sequence. The audio components are recorded individually alongside the background tracks, and their amplitude is maintained constant. All of the audio components (main component and secondary components) are recorded using the same process which is represented in FIG. 5.

FIG. 5 shows a digital recording process 120 with a sequencer/editor 122. Four channels 124, 126, 128, 130 of the digital sequencer/editor 122 are shown as being used. Two of the channels 128, 130, are used for playback of the left (L) and right (R) channels of the background audio track. The other two channels 124, 126, are used for recording audio components and are set up as virtual channels, one for the main audio component and the other for secondary components. Each of the virtual (V) channels is mapped to a folder containing components. V Channel 1 is mapped to a folder containing components 132, 134, 136; while V Channel 2 is mapped to a folder containing components 138, 140, 142.

The main audio component is recorded but with gaps in the recording. When mixed with the secondary audio components during online playback, these gaps are "filled", to generate one acoustically complete audio sequence.

The recording process is represented in FIG. 6 with a simplified block diagram of a voice recording studio 150. A microphone 152, receives audio input which is input to a microphone pre-amplifier 154, which increases signal amplitude and performs dynamic compression of the audio signal. Analog effects within the pre-amplifier can be controlled by an effects processor 156. The analog signal from the pre-amplifier 154 is sampled and converted to a digital signal by means of an analog-to-digital (A/D) converter 158, whose output is stored directly to a disk 160 across a fast communication channel such as Firewire™. Digital data for

background tracks may also be retrieved from the computer/Disk 160 and sent to the mixer 162 through the A/D converter 158.

To record a secondary audio component, the background track and the primary component are retrieved from the computer and played on the mixer. The mixed sequence including the background track, primary track, and secondary track, is played over a microphone monitoring headset to provide feedback, while the secondary track itself is being simultaneously saved on the disk.

Bouncing

Bouncing is the process in which the background track is combined (mixed) with the main (primary) audio component to produce a mixed background track. FIG. 3 represents bouncing 84 as a stage that occurs after the generation of an audio component or a background track. Bouncing is accomplished using the studio setup 150 shown in FIG. 6 wherein the mixer 162 is used for playing back a previously recorded background and mixing it with the primary audio track. The mixer has a set of Left and Right Playback controls 164 for the playback of a previously recorded background track, and a set of Left and Right Microphone controls 166 for monitoring (headsets). The two channels are combined using a Sequencer/Editor.

The following activities take place:

Mixing - a primary audio component is mixed with the background track (L & R) to produce a mixed background track (stored as L & R channels).

Peak limiting - eliminate distortion due to excessive sound amplitudes and optimizing overall sound level.

Audio maximization - enhance low amplitude sound resulting in optimized overall sound level.

Dithering - an algorithm that quantizes (rounds off) the number of digitally stored audio bits to improve sound quality.

Noise shaping – an algorithm which removes inaudible frequencies and noises from a recorded sound, resulting in improved sound quality

Wave Conversion

Wave conversion is an optional process which takes place in the audio sequencer/editor. FIG. 3 represents wave conversion 86 in phantom to show its option nature. When performed, all audio components are converted from CD

audio quality as follows:

	CD Quality	Modified Quality
Sample rate	44.1 kHz	22.05 kHz
Resolution	16 bit	16 bit
Mono/Stereo	Stereo	Mono

Wave conversion reduces the audio file size and may improve sound quality in certain online audio players.

Audio Encoding

The audio components are encoded in any desired encoding scheme, such as WAV, MP3, etc. FIG. 3 represents audio encoding 88, as the final function block before the audio repository 90. Audio encoding takes place as seen in FIG. 6 on the computer 160, utilizing existing format-specific encoders. Considerations for encoding schemes include: file sizes, desired audio quality, and desirable playback technologies. The encoding process compresses the file, thereby reducing the download time.

Examples for audio encoding include:

QuickTime™: *.mov (an MP3 variant)

Beatnik™: *.rmf (an MP3 variant)

Generation of Synchronization Descriptors

The synchronization descriptor file is generated 94 in FIG. 3, so that all activities associated with a particular audio sequence will be synchronized. These activities may include, but are not limited to: invocation of the mixed background track, synchronizing video animation with an audio sequence. Each mixed background track is associated with one unique descriptor file. FIG. 7 shows the synchronization association 170 between a mixed-background track 172 and a synchronization descriptor 174.

The synchronization descriptor 174 as shown in FIG. 7, is a MIDI file generated by the Sequencer/Editor. Synchronization is accomplished through the use of meta-events, which are embedded in the file. For example, using an eMagic sequencer/editor within the embodiment of this invention, allows

embedding up to 128 types of meta-events, with each such meta-event appearing any number of times inside the file. Various meta-events are encoded in the time sequence location as shown in FIG. 7 by the representative markers 176, 178, 180, 182. Marker 176 triggers the playback of the mixed background track, marker 178 triggers the playback of the first secondary component, filling Gap 1 and marker 180 triggers the playback of a subsequent secondary component, filling Gap 2. Marker 182 may trigger the execution of an animation sequence. The synchronization descriptor is encoded and compressed in the same manner that the rest of the audio files are encoded.

Generation of Sequence Descriptors

FIG. 3 depicts sequence descriptor generation 96. The sequence descriptor file defines the order in which audio components are embedded (mixed) within a background track. The sequence descriptor is a text file which is generated manually. The sequence descriptor contains information in the following format:

[DESC] = *audio_component1, audio_component2, audio_component3*

Where:

audio_component1 - is the first audio component to be played.

audio_component2 - is the second audio component to be played.

audio_component3 - is the third audio component to be played.

It should be noted that each of the audio components may be played one or more times within a sequence. For example:

[DESC] = *audio_component1, audio_component2, audio_component1*

Algorithms

Server-Side Algorithms

Composition and transmission of audio sequences on the server-side is represented by the sequence 190 shown in FIG. 8 for the preferred embodiment of the invention. This server-side program used is application specific.

Receive and parse audio sequence request

The audio sequence request 192 is received as an HTML '*post*' command. An example for such a request:

Param1=X&Param2=Y&Param3=Z

The request is then parsed 194 (broken down and interpreted) to produce a set of database retrieval actions. Each of the parameters from within the request is isolated for further use in subsequent steps.

Identify Necessary Files

The parameters are used to identify necessary files 198 which are to be sent to the client application. The audio index database 196 is referred to for specific audio component information. Examples for such information, retrieved from the database:

Identify all music styles with a given sequence length.

Identify all primary audio files within a given music style.

Identify all secondary audio files, which can be played in Gap 1 of a given primary audio component.

Identify the directory path to a given audio component.

Identify the directory path to a synchronization descriptor, associated with a given mixed background track.

The response files from the query comprise audio components and descriptor files. The parameters are translated based on the audio generator rules which are embedded in the program, and assisted by sequence descriptor files and indexing mechanisms, such as logically constructed directory structures or relational databases. For example:

Param1 (X) may define the mixed background track. '*audio_background_7*'

Param2 (Y) may define audio component '*audio_component_4*'

Param3 (Z) may define audio component '*audio_component_2*'

'*Audio_background_7*', selected in step (1), will be sent along with the synchronization descriptor '*midiq_7.rmf*'

'*Audio_background_7*', selected in step (1), will be sent along with the sequence descriptor '*seq_7.txt*'

Send Files To Client Application

The server application sends files to the client application 200. Audio

components necessary for inclusion are read from the audio repository 202. The files sent to the client station are HTML files with embedded audio component files and descriptor files, along with the necessary Java scripts used for invoking plug-ins. The Java scripts are specific to the audio technology (plug-in) being used.

Client-Side Algorithms

Assembly and playback of audio sequences on the client side is represented by the sequence 210 shown in FIG. 9 for the preferred embodiment of the invention. The client-side program is also application specific.

Send Audio Sequence Request

The audio sequence request is sent 212 as an HTML '*post*' command. An example of the request format is given by:

Param1=X&Param2=Y&Param3=Z

Command parameters may be defined through pull-down menus or other interface mechanisms.

Receive Audio Sequence

The audio sequence is received 214 as part of an HTML response. The HTML response comprises embedded audio components, descriptor files, and Java scripts for plug-in activation.

Activate Plug-Ins

The activation of plug-ins 216 is performed in accordance with the Application Programming Interface (API) for browser plug-ins. Depending on the implementation, the client software may activate one or more plug-in instances.

Play Audio Sequence

The audio plug-in plays the audio components 218, 220. The plug-in is responsible for decompressing the audio files, and then playing these in a prescribed sequence. Since the sequence is downloaded in terms of its components, no streaming is being used. The client will complete downloading, before the audio sequence is begins playing.

Mixme.com Example Web Site

This section provides a detailed description of an example web site which employs the principles and methods of the automated generation of customized audio content of the invention. The web site is referred to as "MixMe.com".

The site allows Internet users to compose custom songs and other audio sequences at studio quality levels and send these audio sequences to other Internet users. The site supports the creation of downloadable audio sequences that can be played offline and programming of audio CDs which contain custom sequences, composed online.

Content

The steps involved in the generation of audio content within the MixMe.com site can be represented as a tree hierarchy. Web users select various subject categories, titles and configurable features, such as the recipient's name, hobby and body features, which allows the user to construct a large number of unique audio sequence permutations. The audio sequence consists of a mixed background track along with one or more audio components.

Background Track

Mixed background tracks contain the lyrics of a title (primary component) mixed with instrumentals or any other background sound (laughter, cheering, etc.). The title being sung (or told), and its genre define a background tracks. A variety of audio sequences can be composed by the users, such as:

Music:

Clips - musical sequences of approximately 10 Sec and a single customizable parameter.

Jingles - musical sequences of approximately 30 Sec and 5-8 customizable parameters.

Songs - musical sequences of approximately 3 Min. and 10-15 customizable parameters.

Jokes:

Short jokes - approx. 10 Sec and a single customizable parameter.

Long jokes - approx. 30 Sec and 5-8 customizable parameters.

Announcements:

Announcement - approximately 10 Sec with single customizable parameter.

Users of the site can compose audio sequences on various subjects. A representative sample of the subject categories contained on the MixMe.com site at launch time include: Love, Kewl Krap, Sports, Occasions, and School. The subject categories can be subdivided into sub-categories to any depth desired. Additional categories can be added in the future either statically or by the use of a dynamic add mechanism. Examples of sub-categories for the above categories include:

Love:	Love working, Love not working, Dating, Friendship, Friendship not working
Occasions:	Happy Birthday, Bar/Bat Mitzvah, Graduation, Wedding, Anniversary

The subject sub-categories may contain one or more titles. Representative examples under Love working include the items "Let's cuddle", "Sweet dreams", and "High on you". The sub-category of Happy Birthday could include the titles "Sweet Sixteen", "You're 13", "I'm down with you", and "Rock all night". Obviously the items above are just a few representations of the numerous titles that could be used within the preferred embodiment.

Users can send their audio sequences in various genres. Examples of a few representative genres could include such items under Music as "Rock/Pop", "Hip Hop", "Dance/Pop", "Rave", and "Grunge". Under Jokes a variety of genres may exist such as "New York comic", "Sassy Sally", and others. Announcements can be broken down into various forms of announcer genres, such as "Stadium announcer", "Broadcaster", and so forth. Again, as with the other categories, additional genres and categories can be added, either statically or dynamically.

Audio Components

Secondary audio components contain various parameters, used to customize the audio sequence. Parameters can include such information as the recipient's name, hobbies, and so forth. The secondary audio components are sent as individual files, making up secondary sound tracks, which are mixed with the background track at the client station to form a contiguous audio sequence.

The number of parameters, and their selection menu, depend on the actual title (defined by sequence type, category, sub-category and title). The sequence

descriptor defines the order in which these components are played.

Functional Components

The system provides the following key functions:

(1) Provide the sender with a mechanism for selecting a sequence type, subject category and sub-category.

(2) Provide the sender with a mechanism to compose an audio sequence by selecting from a set of pull down-menus any number of parameters, valid for the previous selection.

(3) Provide the sender with a mechanism for adding a personal text message.

(4) Provide the sender with a mechanism for entering his/her information as well as the recipient's information (e.g.: e-mail address).

(5) Allow the sender to preview the custom audio sequence as well as the graphic presentation associated with the sequence. The sender can either edit the sequence, modifying any of the previously selected/configured items, or send it.

(6) Provide the recipient with a mechanism for listening to the audio sequence as well as view the graphic presentation associated with the sequence.

The system provides additional functions such as e-commerce capability for purchasing custom audio sequences in downloadable format or CDs, members club, corporate information, etc.

System Architecture

The MixMe.com web site comprises a web server that runs an ASP custom application and communicates with a SQL Server™ database.

The ASP program is responsible for the dynamic display of the client's pages, based on the system's rules, retrieval and transmission of the audio files and the creation and transmission of Java scripts that activate plug-ins on the client station.

The database supports the ASP program and maintains the content rules as entity relationships. It also functions as an index to the audio repository

The client station consists of an Internet browser and plug-ins, used for graphic display and audio playback. Plug-ins include:

Macromedia Flash™ and ShockWave™

Beatnik™ audio player

FIG. 10 is a network diagram of the internet system 230 topography and connectivity to support the MixMe.com web site. The web site is hosted within an Internet Service Provider (ISP) 232 environment and consists of a fully redundant setup. Connectivity between the web site and the Internet is facilitated by a router 234. The router delivers all network packets, targeted for the site to its destination. Similarly, traffic, originating in the site and targeted to remote client station is sent to appropriate network paths. All of the computers which make up the site reside on a 10/100 Mbps Local Area Network (LAN). Connectivity between the various nodes on the LAN is facilitated through a 10/100 Mbps switch 236. To achieve a higher degree of fault resilience, a separate power switch 238 powers the network switch. To achieve a high degree of scalability as well as fault resilience, two or more web servers, exemplified in Fig. 10 by 244, 246 and 248 form a redundant, load-sharing cluster, with traffic being load-balanced by a BigIP™ switch 240. The BigIP switch functions also as a packet validator. A cluster of database servers 250 consist of two or more servers 252 and 254. It is linked to the cluster of web servers and hosts the audio index database. To eliminate a single point of failure at the load balancing switch, a failover BigIP switch 242 is being provided.

Song Selection and Configuration

Referring to FIG. 11, a logic diagram 260 of selection rules for the audio components is shown. The diagram displays the entity relationship between the selectable audio parameters 262, namely: sequence type (e.g. clip, jingle, short and long jokes, announcement), a subject category (e.g.: love, school, etc.), subject sub-category (e.g.: love working, love not working, etc.), individual titles, genre (e.g.: Rave, Rock, Stadium announcement, etc.), and the selected name and parameters. A one-to-many relationship exists between the Sequence Types and Category 280, Category and Sub-Category 282, and Sub-Category and Title 284. Additionally a one to many relationship 286 exists between Sequence Types and Genre. A set <sequence type, category, sub-category and title> defines a set

of selectable parameters 264 that may be used with the audio sequence. A set is shown <title, genre> 266 that defines a background track 268 (primary audio component). A selected genre is mapped 270 to a singer. A set <singer, recipient name> defines 272 an audio name file 274 (secondary audio component). A set <singer, parameter> defines 276 an audio parameter file 278 (secondary audio component). These rules are enforced through the server-side application program and the database.

Server Design

The audio server architecture was described in the section entitled Audio Application Server. The audio server program is responsible for:

(a) Audio sequence generation

(b) User interface with the audio generation engine (composition pages)

(c) Image engine – dynamic display of images

(d) Presentation engine – coordination of audio composition and associated pages and images.

(e) User notification via e-mail messages

Audio Sequence Generation

The system parses the input form, selects the appropriate audio components through the use of an audio database as described above and transmits them to the client station along with a custom Java script which activates the Beatnik player.

User Interface

The sender is led through four composition steps:

(a) Selection of sequence type (music [short, medium, long], jokes [short, long] or announcements).

(b) Selection of a subject category and sub-category.

(c) Selection of a style (e.g.: Rock, R&B music, stand-up comedy, stadium announcement, etc.).

(d) Selection of appropriate parameters and recipient name.

The steps of the composition process are implemented as a state machine with each of the four composition pages displayed dynamically in accordance with a set of composition rules in relation to the available repertoire of audio components which are suitable for the user's selection.

Image and Presentation Engine

Each of the web pages within the composition and review process is generated dynamically and associated with context specific graphics. The server is responsible for selecting the appropriate graphics for the current context.

Following are two examples for content-specific graphics:

(a) Subject categories are displayed dynamically. Each subject has an associated icon ('hearts' for Love, 'books' for School).

(b) Background animation for a Rock selection will include guitars, while animation for an 'India Vibe' selection includes an Indian goddess.

Dynamic (and context sensitive) display of the graphics is accomplished through an ASP program, and through the use of a SQL relational database.

User Notification

The recipient is notified via e-mail about an audio sequence that was created for him/her. Within the preferred embodiment of the invention, the recipient then hits a hyperlink within the e-mail to access the MixMe.com web site to retrieve the message. The response link is coded with an identification that is looked up in a database within the Mixme site to reference a recipient audio file descriptor which contains audio sequence pointers from which the recipient's message may be constructed. The use of these descriptors reduces the storage requirements within the Mixme.com site since the recipient audio file descriptor contains pointers instead of actual audio files. The recipient audio file descriptor is coded similarly to the sequence descriptors previously defined. The sender originating the message is notified via e-mail once the audio sequence is "picked up" by the recipient. Notification via e-mail is implemented through an ASP program and ASP-mail COM modules.

Alternate mechanisms exist by which the user can be notified. One alternative is to sent the recipient the actual recipient audio file descriptor for playing of the audio file composed by the user. These descriptors can be used on the Mixme.com site to access the audio component database so that the message may then be played back. Sending descriptors to a recipient requires less download time and storage than required for actual audio files. The user may be sent the actual composed audio file for local playback as well.

Client Design

The client side architecture was described in the previous section entitled Audio Client Architecture. The client program uses a standard Internet browser and several plug-ins. It displays HTML pages, sent by the server program. The client program has two primary functions:

(a) Select/compose an audio sequence.

(b) Listen to a previously composed audio sequence.

Selection/Composition Of An Audio Sequence

The following example program generates an HTML form, with pull-down menus for the selection of a recipient name, his/her hobby, characteristics and body features. Additional pull-down menus allow the selection of music genre and title.

```
<form method="post" action="compose.asp" name="makeSForm"
target="player">

Edit The following elements and click preview

<!-- Select a recipient name -->
<select name="theName">
      <option value="Abe">Abe</option>
      <option value="Bob">Bob</option>
      <option value="Dave">Dave</option>
</select>

<!-- Select a hobby -->
<select name="Hobby">
      <option value="Ballet-Dancing">Ballet Dancing</option>
      <option value="Biking">Biking</option>
      <option value="Boogie-Boarding">Boogie Boarding</option>
      <option value="Bowling">Bowling</option>
      <option value="Boxing">Boxing</option>
      <option value="Bungie-Jumping">Bungie Jumping</option>
</select>


<!-- Select a characteristic -->
<select name="Char">
      <option value="Amazing">Amazing</option>
      <option value="Beautiful">Beautiful</option>
```

```
            <option value="Breath-Taking">Breath Taking</option>
            <option value="Classy">Classy</option>
            <option value="Dazzling">Dazzling</option>
      </select>

      <!-- Select a feature -->
      <select name="Feat">
            <option value="Black-Hair">Black Hair</option>
            <option value="Blonde-Hair">Blonde Hair</option>
            <option value="Curly-Hair">Curly Hair</option>
            <option value="Blue-Eyes">Blue Eyes</option>
      </select>

      <!-- Select a music genre -->
      <select name="SongStyle">
            <option value="Hiphop">Hiphop</option>
            <option value="Blues">Blues</option>
            <option value="Rock">Rock</option>
      </select>

      <!-- Select a title -->
      <select name="SongNumber">
      <option value="1">Love Working</option>
      <option value="2">Personal Appearance</option>
      <option value="3">Sports</option>
      <option value="4">Happy Birthday</option>
      <option value="5">School</option>
      </select>

      </form>
```

When run on the client browser, the HTML program above results in the browser screen 290 shown in FIG. 12 with the user selection fields Name 292, Hobby 294, Characteristic 296, Features 298, Music Style 300, and Lyrics Theme 302.

Playback Of An Audio Sequence

The client program uploads a request to the web server of the MixMe.com site for an audio sequence, and subsequently downloads a response from the server. The response contains the audio components, (encoded MIDI) a

synchronization file, and Java scripts for activation of the plug-ins. The audio components are loaded into an ordered array.

The client application employs multiple instances of the Beatnik plug-in (player), one for each audio component. Additionally, a MIDI synchronization file (midiq.rmf) is played on one of the Beatnik players as well.

One of the players is started and plays the synchronization file. This player sets the pace and event triggers used by the rest of the players. The client application monitors for meta-events which are embedded in the synchronization file, and starts an activity with each meta-event. Meta-events of type "marker", trigger the activation of a Beatnik player, playing an audio component marked as current. Other meta-events can trigger additional activities which are synchronized with the audio sequence. An example of one such event is the scrolling of sections of lyrics, which are triggered by meta-events of type "genericText". An example of an event for an HTML file, sent as a response to the previous request is shown below:

```
<SCRIPT LANGUAGE=JavaScript><!-- //
new Music ('midiPart');
new Music ('mainPart');

// create the parts array
var partArr = new Array();

// Open as many players as needed
      new Music ('part1');
      partArr[0] = 1;
      new Music ('part2');
      partArr[1] = 2;
      partArr[2] = 1;
      new Music ('part4');
      partArr[3] = 4;
      new Music ('part5');
      partArr[4] = 5;
      partArr[5] = 1;
state=-1;

function startPart(partNum) {
 //   alert (partNum);
 //     alert('part' + partNum + '.play();');
 //     if (partNum == 1) {
 //          document.myMovie.EvalScript(3)
 //     }
```

```
        eval('part' + partNum + '.play();');
 //      part1.play();
}

function MetaCalled(metaEventType,metaEventValue) {
        if (metaEventType == "Marker") {
                if (state == -1) {
                        mainPart.play();
                        state = state+1;
                } else {
                        startPart (partArr[state++]);
                }
        } // Marker

        if (metaEventType == "GenericText") {
                document.myMovie.EvalScript(songLines[curLine++]);
        }
}

function StartAll() {
        document.myMovie.EvalScript(0);
        midiPart.stop();
        // make sure midi is in first position
        midiPart.setStartTime(0);
        midiPart.play();
}

function stopPlayers() {
        if (state > -1) {
                midiPart.stop();
                mainPart.stop();
                //      if (state > 0) {
                //         state1 = state - 1;
                //         eval('part' + state1 + '.stop();');
                //      }
        }
        state=-1;
        curLine = 0;
}

function playBeat() {
        stopPlayers();
        StartAll();
}

function stopBeat() {
```

```
        stopPlayers();
    }

    midiPart.onMetaEvent ( MetaCalled);
    mainPart.onLoad(StartAll);
    // --></SCRIPT>


    <script language="JavaScript"><!-- //
        part1.preloadEmbed('../DCR/Name/Abe.rmf');
        part2.preloadEmbed('../DCR/Hobby/Biking.rmf');
        part4.preloadEmbed('../DCR/Char/Classy.rmf');
        part5.preloadEmbed('../DCR/Feature/Black-Hair.rmf');
        mainPart.preloadEmbed
    ('../DCR/songLy/Hiphop/Song1/main.rmf');
        midiPart.magicEmbed ('SRC="../DCR/songLy/Hiphop/midiQ.rmf
        TYPE="audio/rmf"  AUTOSTART="FALSE"  WIDTH=148 HEIGHT=45');

        // make sure that midi does not stop till main is loaded
        midiPart.stop();
    // --></script>
```

In this example, the mixed background, main.rmf, corresponds to a Hip-Hop jingle used in celebrating a birthday. In addition to the mixed background, four other audio components are downloaded into an array. These audio components correspond to:

Name = 'Abe'

Hobby = 'Biking'

Characteristics = 'Classy'

Features = 'Black hair'

Additionally, the system downloaded the synchronization file, midiQ.rmf, which corresponds to Hip-Hop jingles.

midiPart.play() is responsible for starting the synchronization descriptor.

mainPart.play() is responsible for starting the mixed background track.

StartPart() is responsible for starting all secondary audio components.

MetaCalled() is responsible for monitoring for meta-events and initiating the appropriate activity.

Data Structures

Directory Structure (audio repository)

The audio repository is stored as a file directory structure on the web server and is indexed by an SQL database that enforces the audio generation rules. The audio repository can be thought of as a pair of primary components, a main audio component which are primarily background tracks, and secondary components which are audio parameters and name files. Although the database provides full transparency, the directories follow a logical structure. The logical structure used in the preferred embodiment simplifies content maintenance and is consistent with the recording and composition logic. FIG. 13 shows a representative structure for the sequences within the audio repository, while FIG. 14 shows a representative structure for the parameters within the audio repository.

A sequence tree 310 is shown in FIG. 13, where the tree branches out from the origin 312 of the audio repository sequence tree, to various directories, each corresponding to sequence types, of which Clip 314, Joke 316, and Announcement 318 are shown as a representative sample. Each one of these sequence type directories is further categorized in audio types, each of which resides in a separate directory. Audio types under Clips 314, are represented by Rock 320, Jazz 322, and Hiphop 324. Categorized under each audio type are audio sequence titles, each of which resides in a separate directory. A few such titles are shown under the Jazz type 322, as "Title 1" 326, "Title 2" 328, and "Title N" 330. Additionally categorized under the audio types are synchronization descriptors as represented by midiq.rmf 332. Each audio type directory contains a single midiq.rmf file, common to all files within the subordinate branches. Under each Title directory is stored the sequence element for that title, as shown by the set of files called main.rmf 334, 336, 338.

In similar manner to the sequence tree, a parameter tree 340 is shown in FIG. 14, where the tree branches out from the origin 342 of the audio repository parameter tree, to various parameter types. This tree provides the choice lists that allow the user to customize his sequences by selecting parameters. Under the main parameter tree 342 are parameter categories, represented by Name 344, Parameter m (Hobby) 346, and Parameter n 348, each of which resides in a separate directory. To further show the underlying structures the Name 344

parameter is divided into Name_Variant_1 350, Name_Variant_2 352, and Name_Variant_n 354, each of which resides in a separate directory. These directories correspond to name recordings by an individual singer. Representing the specific name selection under Name_Variant_2 352 are a set of audio name parameter files name_1.rmf 356, name_2.rmf 358, name_n.rmf 360. Similarly another parameter, Hobby, 346 is shown with a few representative variants Hobby_Variant_1 362, Hobby_Variant_2 364, Hobby_Variant_n 366, each of which resides in a separate directory. Each of these variants corresponds to hobby recordings by an individual singer. Each of these variants contains specific audio parameter selections which in this case are represented by Hobby_1.rmf 368, Hobby_2.rmf 370, and Hobby_n.rmf 372.

Database Schema

A database schema for the preferred embodiment instantiated as the MixMe.com web site is shown in FIG. 15 and FIG. 16. The schemas depict relationships between tables within the database used for MixMe.com. A complex web site, such as MixMe.com, that provides a wide variety of services will require a large database for holding and relating the elements therein. The schema shown in FIG. 15 and FIG. 16 is shown to depict some major elements within such a database and the relationships therein. A person with ordinary skill in the art can create extend this schema or create a new one of any arbitrary size.

The representative schema 380 of FIG. 15 contains named tables. Each table has a name (the top line) and a set of fields.

The *PlaceTypes* table 382 contains a key field *pkPiece* and a string field *strPieceName*.

The *Categories* table 384 describes subject categories. It contains a key field *pkCat* as well as the fields *fkPiece1*, *fkImageSet1* and *strCategory*.

The *SubCategories* table 386 describes subject sub-categories. It contains a key field *pkSubCat* as well as the fields *fkCat1*, *fkBkgndSeq* and *strSubCategory*.

The *BkgndSeq* table 388 describes a background sequence. It contains a key field *pkBkgndSeq* as well as the fields *strBkgndmidiP*, *Param* and *Seq*.

The *Parameters* table 390 describes parameter groups. It contains a key field *pkParam* as well as the fields *fkSinger2*, *strParameter* and *fkBgnSeq*.

The *SubParameters* table 392 describes sub-parameters. It contains a key field *pkSubParam1* as well as the fields *fkParam1* and *strSubParameter*.

The *Styles* table 394 describes music styles. It contains a key field *pkStyle* as well as the field *fkSubCat2*.

The *Singers* table 396 describes singers. It contains a key field *pkSinger* as well as the fields *fkStyle, strSingerName, strGender* and *strEthnicity*.

The *ParameterBytes* table 398 describe the parameter audio files. It contains a key field *pkParamByte*, as well as the fields *strParamByteP*, *fkSinger3*, *strByteName*, *fkSubParam2* and *bName*.

The *BackgndOrdByte* table 406 defines a descriptor file. It contains the key field *pkBkgndOrdByte* as well as the fields *fkBkgndTempl*, *fkParamByte1*, *strByteText*, *iOrder* and *fkSubParam2*.

The *ClientLib* table 408 describes the client's browser software and contains the key field *pkClientLib* as well as the fields *strName*, *strLibP*, *iMajorVer*, *iMinorVer* and *iRev*.

The *BackgroundBytes* table 400 describes the background track. It contains the key field *pkBkgndP* as well as the fields *strBkgndP*, *strBkgndName*, *fkSinger3* and *iChecksum*.

The *BackgndTemplates* table 402 defines the synchronization descriptor. It contains the key field *pkBkgndTempl* as well as the fields *strMidiP* and *fkBkgndByte*.

The *BkgndOrdLyric* table 404 defines the ordering of the parameters within the background track as well as the lyrics sung. It contains the key field *pkBkgndOrdTemplTxt* as well as the fields *fkBkgndTempl2*, *strTemplText* and *iOrder*.

The representative schema 410 of FIG. 16 contains named tables. Each table has a name (the top line) and a set of fields.

The *UserToMix* table 412 correlates user information to mixes, waiting to be picked up. It contains a key field *fkMix1* and a field *fkUser1*.

The *Users* table 414 stores information about a single site visitor. It contains a key field *pkUser* as well as the fields *strFName*, *strLName*, *strEMail* and *strType*.

The *Mixes* table 416 stores mix information, for mixes waiting to be picked up. It contains a key field *pkMix* as well as the field *strMix*.

The *Log* table 418 maintains a log of all visits. It contains a key field *pkLog* as well as the fields *iLevel*, *strSource*, *iNum*, *dTimeStamp* and *strDesc*.

The *ImageSets* table 420 describes image sets being displayed on the client station. It contains the key field *pkImageSet* as well as the field *strImageSetName*.

The *Images* table 422 describes images being displayed on the client station. I t contains a key field *pkImage* as well as the fields *strImageP*, *fkImageSet1*, *iWidth*, *iHeight*, *strAltTag* and *bFlash*.

Site Architecture

The MixMe.com web site consists of two main parts. One part utilizes Flash technology, while the other part is intended for users that do not have a Flash plug-in. The site architecture of FIG. 17A through FIG. 18B, shows representative navigation paths through the Mixme.com web site. FIG. 17A represents the Flash site and the entry point for both sites 430, while FIG. 18A represents the non-Flash site 490.

The representations of FIG. 17A through FIG. 18B are simplified block diagrams of the high-level functional flow within the preferred embodiment of the invention. Each block within FIG. 17A through FIG. 18B, contains a page name or state within a particular page and contains an associated page address. Within these block diagrams, the use of "Back" or the on screen selection that results in going to a previous screen has not been represented so that the normal flow of decisions can be more clearly shown. The highest level decisions within the block diagrams are represented within a diamond shaped decision block; however the rectangular blocks which represent screens or screen states also provide for selections (decisions) that can cause a transition to another block.

In FIG. 17A the sender enters the site 432 and goes into the Sensing Page 434 in which the type, version and capabilities of his browser are sensed. If a Flash plug-in is detected 436, the user is automatically directed to the introduction screen of the Flash site 440, otherwise the user is automatically directed to the non-Flash site (Basic site) of FIG. 18A via off-page connector "C" 438. The Sensing Page has no display associated with it.

Flash Site

The Flash Intro page 440 displays an animation associated with the MixMe site and then the screen is refreshed with the user directed to a Splash Page. The system accesses the user cookie on the client station, to determine if this is a first time user 442. For a first-time user, or a user whose cookie can not be accessed, a sign-on Splash page 444 is displayed, prompting the user to enter their name. Repeat users skip the sign-on step since the name information is embedded in the cookie. In either case, the user is taken to a Splash Page 446 in which a brief custom introductory song is being played.

At the end of the introductory song, the system checks for cookies on the client side 448, and if none are found, a cookie will be downloaded from a designated page 450. Once the system has ensured that the client station has the appropriate cookie, the user's browser will be directed to the MixMe Home Page which is shown in FIG. 17B via off-page connector "F" 452.

The MixMe Home Page 484 of FIG. 17B offers the user the ability to select a pre-defined mix or to start with the composition process. If the user at the Homepage elects to compose a message, then the Composer page 486 is entered. The Composer page consists of four distinct states, or steps. Step 1 of the composition process 486 allows the user to select a Mix type (music, jokes or announcement). Composer Step 2 488 allows the user to select subject categories and sub-categories (e.g.: Love Working, Occasion – Wedding, etc.). Composer Step 3 490 allows the user to select a Mix style (e.g.: Rock, Rave, etc.). Composer Step 4 492 allows the user to select various parameters and the recipient name. While in the composer page the user has the opportunity to go back and modify previous selections (not shown). Once the user considers his/her selection complete, they may select to preview the message. Before entering the Preview Page 500 a check for the Beatnik plug-in 494 is performed. If the plug-in is not installed 496, then the user is allowed to download it. If they elect not to download it 498, then they will not be able to play audio content and will be given alternatives or ushered back to the main menu. If the user does download the plug-in, another check on the plug-in is made 494, and if properly installed then the preview screen 500 is entered.

The Preview page 500 provides a preview of the recipient's message, including the audio sequence along with any associated animation and textual messages. Once the playback is completed, the sender has the option of returning to the composer pages to edit the existing selection (not shown), or moving on to the Send Page 502. When sending an audio message, the user enters a recipient name and e-mail address. Following the transmission of the message, the user is directed to a Thank You Page 504, where they are invited to send another audio message. The recipients ISP will then receive the e-mail and the recipient will be notified.

If the user selects a predefined mix from the Home Page 484, then one of the composition pages 486, 488, 490, 492, will be entered. Which of the composition screens is entered depends on what remains to be defined within the predefined mix.

In response to the e-mail notification the recipient enters the site 454 in FIG. 17A shown as connector "B", with a custom target audio sequence. The browser senses for Flash mode in the Sensing Page 456. Once it is determined if the recipient's browser supports flash 458, the recipient continues on within the Flash site, or is transferred to the Basic site if Flash is not supported. Program flow to the Basic site is shown via connector "D" 460. Within the Flash system, it will be verified that the user has a Beatnik plug-in 462, after which the recipient is directed to the Recipient Page 468, where he/she can listen to the audio sequence. Recipients that do not have a Beatnik plug-in will be automatically directed to an Installation Page 464 where Beatnik can be downloaded and installed. After listening to the audio message, the recipient is invited to either reply with an audio message by entering the composer 450, or to go to the home page shown in FIG. 17B via off-page connector "G" 470.

Non-Flash Site 490

The Non-Flash site is represented in FIG. 18A and FIG. 18B. The sender enters the basic site, non-Flash site, 510 once it has been determined that his/her browser doesn't support Flash. Entry of a sender is shown by off-page connector "C" 512. The system determines 514 whether this is a first-time user or a repeat user, based on a cookie which it attempts to find on the client station. For first-time users, or those without a valid cookie, a Splash sign-on page 516 is

displayed, prompting the user to enter their name. Repeat users with a valid cookie skip the sign-on step since the system automatically detects the name of the user which is embedded in the cookie. In either case, the user is taken to a Splash Page 518 in which a brief custom introductory song is played. At the end of the introductory song the system checks for cookies on the client side 520 and if none are found, a cookie will be downloaded from a designated page 522. Once the system has ensured that the client station has the appropriate cookie, the user's browser will be directed to the MixMe Home Page shown in FIG. 18B via off-page connector "J" 524.

The Home page section of the non-Flash site is shown 540 of FIG. 18B. Senders arriving at the MixMe Home Page 542 are allowed to select a pre-defined mix or to start with the composition (configuration) process. The Configurator page consists of four distinct states, or steps. Step 1 of the Configurator Page 544 allows the user to select a Mix type (music, jokes or announcement). Step 2 of the Configurator Page 546 allows the user to select subject categories and sub-categories (e.g.: Love Working, Occasion – Wedding, etc.). Step 3 of the Configurator Page 548 allows the user to select a Mix style (e.g.: Rock, Rave, etc.). Step 4 of the Configurator Page 550 allows the user to select various parameters and the recipient name. While in the configurator pages the user has the opportunity to go back and modify previous selections. Once the user has completed making selections they may select to preview the message. Before entering the Preview Page 558 a check for the Beatnik plug-in 552 is made, if the plug-in is not present, the user may elect to download it 554. If they elect not to download it 556, then they will not be able to play audio content and will be given alternatives or ushered back to the main menu. If the user downloads the plug-in another plug-in check is made 552, and if the plug-in is properly installed the preview page 558 is entered.

The Preview page 558 provides a preview of the recipient's message, including the audio sequence along with any associated animation and textual messages.

Once playback is completed, the sender has the option of returning to the composer pages and editing the existing selection, or sending it. When sending an audio message from the Send Page 560 the user enters a recipient name and

e-mail address. Following message transmission, the user is directed to a Thank You Page 562, where they are invited to send another audio message.

If the user selects a predefined mix from the Home Page 542, then one of the composition pages 544, 546, 548, 550 will be entered. Which of the composition screens is entered depends on what remains to be defined within the predefined mix.

In response to an e-mail notification from a sender, a recipient without Flash mode enters the site 454 as described in FIG 17A, and is routed to the entry point "D" 526 of FIG. 18A which contains the Basic system. Within the Basic system it will be verified that the user has a Beatnik plug-in 528, after which the recipient is directed to the Recipient Page 534, where he/she can listen to the audio sequence. Recipients that do not have a Beatnik plug-in will be automatically directed to an Installation Page 530 where the Beatnik plug-in can be downloaded and installed. After listening to the audio message, the recipient is invited to either reply with an audio message by entering the configurator via off-page connector "K" 536 to FIG. 18B. The user may elect to transfer to the home page 542 of FIG. 18B.

Page Architecture

Instances of the online composition and playback system are created as web sites such as the Mixme.com site. Each such site comprises a collection of web pages wherein the user navigates in order to perform the various functions provided by the system. The following describes attributes of pages used within the Mixme.com site. It should be recognized that each instance of the online composition and playback system of the present invention may provide the features of the invention within a site that is organized differently than the Mixme.com site.

Home Page

Graphical content -

The home page has three predefined boundaries consisting of the Logo Box, Navigation Box, and TV Box. The definitions for each box are described below.

Logo Box: This element comprises either a movie for the flash site or an image for the basic site.

Navigation Box: This element comprises two possible states based on the number of times the client has visited the site. This section is HTML/ASP based therefore limiting the design to images only.

While in the first state (1st time user), contains a form that prompts the user for his/her name. There are only two required elements for this boundary. An input text box and a submit button. When the 'submit' button is pressed the screen refreshes to the second state.

The second state is used when the name has already been defined and consists of an image. This page plays also a *.wav file containing the name previously entered.

TV Box: This element comprises a flash file.

Musical content - Background music

Customized "clip" (e.g.: "Hello Monique, welcome to our site")

Textual content - Brief intro, inviting users to try the capabilities of the site. (e.g.: "Get Into the Mix! Type in your name to hear a custom song starring ... you! Type your name here _____")

Entry Page

Graphical content -

Logo Box: This box comprises a flash file

Navigation Box: Same as the logo box.

TV Box: Most of the audio sequence selection, composition and preview activity occurs this boundary. In order to create a dynamic flash movie a design was chosen with the ability to dynamically interchange certain elements within the template movie. These parts include:

TV picture/animation

Three icons on the right side of the TV picture

A movie associated with the event for each icon (a total of three "event movies")

There are a total of eight movies, including the template movie, seven of which can be changed periodically by an administrator.

Musical content - None

Textual content - Branding text
Description of the easy song composition steps
Rollover text, describing the buttons

Composition Page, Step 1

The composition pages provides composition functions that allow the user to create the customized audio messages.

Graphical content -

Logo Box: This box comprises a flash file
Navigation Box: Same as the logo box.
TV Box: Flash file
Mix buttons (e.g.: Music, Jokes, Announcements)
(Global) navigation buttons (e.g.: about us, becoming a member, etc.)

Musical content - None

Textual content - Rollover text, describing the buttons

Composition Page, Step 2

Graphical content -

Logo Box: This box comprises a flash file
Navigation Box: Same as the logo box.
TV Box: Flash file
Subject selection buttons (categories, subcategories)

(Global) navigation buttons (e.g.: about us, becoming a member, etc.)

Musical content - None

Textual content - Rollover text, describing the buttons

Composition Page, Step 3

Graphical content -

Logo Box: This box comprises a flash file

Navigation Box: Same as the logo box.

TV Box: HTML form file

Sub-parameter selection buttons (e.g.: hobby/habit; sub-category: wild & crazy)

(Global) navigation buttons (e.g.: about us, becoming a member, etc.)

Music style buttons/icons

Preview button

Musical content - Background audio for music selection buttons

Textual content -

Parameter names (e.g.: Hobby/Habit, Features, etc.)

Pull-down menus (HTML forms)

Free-form text entry field (personal message) – HTML form

Rollover text, describing the buttons

Audio Preview Page

Graphical content -

Logo Box: This box comprises a flash file

Navigation Box: Same as the logo box.

TV Box: HTML or Flash file

'Edit' & 'Send' buttons

(Global) navigation buttons (e.g.: about us, becoming a member, etc.)

Musical content - Playback of audio sequence

Textual content -

Rollover text, describing the buttons

Personal message

Lyrics

Recipient Page

Graphical content -

Logo Box: This box comprises a flash file

Navigation Box: Same as the logo box.

TV Box: HTML or Flash file

Download buttons for Beatnik, other plug-ins

(Global) navigation buttons (e.g.: about us, becoming a member, etc.)

Musical content - Playback of audio sequence

Textual content -

Invitation (e.g.: "Have we got a mix for you...")

Browser and plug-in requirements

Terms of use

Play

Textual content -

Invitation to send a Mix (e.g.: "Why not send (sender's name) a Mix?")

Invitation to become a (free) member

Invitation to buy a downloadable/CD version of the Mix

Thank You Page

Graphical content -

Logo Box: This box comprises a flash file

Navigation Box: Same as the logo box.

TV Box: Flash file

(Global) navigation buttons (e.g.: about us, becoming a member, etc.)

Promotional graphics (e.g.: banners)

Musical content - None

Textual content –

Confirmation that Mix will be sent (description of Mix, recipient)

Invitation to send another Mix

Invitation to become a (free) member

Invitation to browse through the site

Roll-over text describing buttons, etc.

Occasions Page

Graphical content -

Logo Box: This box comprises a flash file

Navigation Box: Same as the logo box.

TV Box: Flash file

(Global) navigation buttons (e.g.: about us, becoming a member, etc.)

Selection buttons (B-Day, Anniversary, etc.)

Promotional graphics (e.g.: banners)

Musical content - None

Textual content –

Roll-over text describing buttons, etc.

Page Layout

A representative example of a selection/composition page 570 within the MixMe.com site is shown in FIG. 19. The user makes selections according to the choices on the menu 574 which is shown on a computer screen 572. The choices shown include Name 576, Parameter 1 578, Lyric 580, and a reserved field 582.

Description of Variations and Alternate Embodiments

It will be appreciated that the invention can be implemented in a variety of ways while adhering to the teachings of the inventive principles. In particular, the following is a partial list of variations that anyone skilled in the art could implement without creative inspiration.

The largest network currently in place for user to user networking is the internet. Implementation of the online composition and playback system is therefore described in relation to this generic networking medium. However the system may be employed within any set of Client-Server applications that reside on private networks, LANs or WANs. Generally within these networks a variety of computers (i.e. PCs, Macs, Servers, and stations) comprise the network nodes. However it should also be realized that the system may be employed within

networks wherein non-computer Internet clients, generally known as Internet appliances, are employed. An ever increasing number of these internet appliances are being developed such as set-top boxes, MP3 recorders, intelligent instruments, cash-registers, PDAs, and so on. Custom audio sequences can be generated online using a variety of Internet technologies. The system can be implemented with various Internet Web Servers as a host, while the client side application can be made to operate with various internet clients (browsers). Additionally the system may be implemented with a variety of third-party or application-specific components that include Internet servers, Internet clients and audio players.

The online composition and playback system described will operate with any form of database. The choice of using an SQL database for storing indexes, coupled with a flag-file audio repository is used within the preferred embodiment to provide an efficient mechanism with easily implemented file structures. Although performance may suffer, the SQL database itself can be eliminated and all data within the system may be contained within a single file system. The data base choices are therefore non-restricted wherein various relational, flat-file, and other database systems can be used for storage of the audio files and descriptor information.

The content for the online composition and playback system may be created in a number of ways. The preferred embodiment describes a recording studio environment and setup used an as example of how tracks and audio components can be recorded. Today an ever-increasing array of methods exist for capturing or collecting audio content that may be employed within the system. Audio content, which includes music, spoken audio in various languages, and sound effects, may be collected through the use of third-party recordings (artist's albums, etc.), uploaded audio (e.g.: karaoke), third party audio synthesizer packages, to name just a few. Additionally the text describes mechanisms that may be used for post processing of recorded audio segments. The forms of post-processing available is extensive and an assortment of these techniques including but not limited to wave shaping, noise Custom, studio-recorded audio reduction, sound effects generation, and compression, which may be employed within the inventive system for enhancing audio content for use on the system. The

collected content may itself be mixed in various configurations, one example of which is using a background track that includes all instrumentals but no vocals. Another variation would provide for the downloading each instrument as a separate track.

Internet technology is progressing rapidly with various encoding techniques along with associated plug-ins being introduced rapidly. The online composition and playback system of the present invention can be implemented to use any of a variety of these encoding techniques and plug-ins for audio playback. The plug-ins chosen within the description of the preferred embodiment were but one form that currently provides efficient audio storage and playback. These encoding techniques may include various forms of audio download mechanisms such as: multi-channel, single channel, streaming and non-streaming.

Although example Java scripts and sections of HTML code are shown within the description of the preferred embodiment, it will be obvious to those skilled in the art that a variety of programming technologies and languages may be used to accomplish the programmed operation described. A partial list of those technologies and languages that may be used includes, but is not limited to: HTML, XML, CGI, ASP, Java, Java Script, VBScript, C and C++.

The preferred embodiment of the online composition and playback system describes use on an example site referred to as Mixme.com. It will be obvious to anyone skilled in the art of site design that the system and principles described herein may be used in sites of various designs and page architectures, wherein the accompanying textual, graphical, and audio content are variables determined by the site creator.

As can be seen, therefore, the present invention provides a system and method for generating custom audio sequences online as a method for sending messages from one network user to another. The method involves pre-recording audio components and performing post-production enhancements of these audio sequences, serving audio components by the use of an application server, accepting and compiling audio components by the use of a client application, and playing custom audio sequences. Audio components are recorded as background and foreground tracks, in a way that when mixed together, they will result in a contiguous, high quality audio sequence. In addition, a large number of audio

sequences can be generated online, through the use of a smaller number of interchangeable audio components. On-line composition tools are provided for generation of custom audio sequences which includes a library of pre-recorded audio components. Additionally, server storage requirements are greatly reduced because the audio sequence is downloaded in terms of its components.

Although the description above contains many specificities, these should not be construed as limiting the scope of the invention but as merely providing illustrations of some of the presently preferred embodiments of this invention. Thus the scope of this invention should be determined by the appended claims and their legal equivalents.

CLAIMS

What is claimed is:

1. A method for online composition and playback of audio content, comprising the steps of:

(a) viewing, using a computer, a listing of selectable audio components for inclusion in an audible composition;

(b) transmitting, using said computer, a selection of audio components for inclusion in an audible composition; and

(c) transmitting, using said computer, a request to generate an audible composition corresponding to selected audio components.

2. A method as recited in claim 1, wherein data parameters for assembly of said audible composition are generated in response to said request.

3. A method as recited in claim 1, further comprising the step of transmitting, using said computer, a request to notify a recipient of the availability of said audible composition for playback.

4. A method as recited in claim 1, further comprising the step of transmitting using a computer an e-mail notification to said recipient that said composition is available for playback.

5. A method as recited in claim 4, further comprising the step of transmitting to said recipient a code which controls assembly of said audio components into an audio sequence corresponding to said composition.

6. A method as recited in claim 1, wherein said audio components are stored on a web server, and wherein said computer comprises a client computer connected to said web server through a communications link.

7. A method as recited in claim 1, wherein said selected audio components are assembled to compose audio content for playback using web browser software.

8. A method as recited in claim 1, wherein said data files corresponding to said composition are stored on a web server.

9. A method as recited in claim 8, further comprising the step of said recipient transmitting to said web server, using a client computer, a code for assembling said selected audio components into said composition.

10. A method as recited in claim 1, further comprising the step of transmitting, using said computer, a sequence for said selected audio components to be assembled into said composition.

11. A method as recited in claim 10, wherein said composition comprises contiguous audio corresponding to said selected audio components and said sequence.

12. A method as recited in claim 1, wherein said composition comprises contiguous audio generated from said audio components.

13. A method as recited in claim 1, wherein said composition comprises a dynamic audio sequence generated from said audio components.

14. A method as recited in claim 1, wherein said audio components are mixable as a time-based sequence.

15. A method for online composition and playback of audio content, comprising the steps of:

(a) providing, using a computer, a listing of selectable audio components for inclusion in an audible composition;

(b) receiving, using said computer, a selection of audio components for inclusion in an audible composition; and

(c) receiving, using said computer, a request to generate an audible composition corresponding to selected audio components.

16. A method as recited in claim 15, wherein data parameters for assembly of said audible composition are generated in response to said request.

17. A method as recited in claim 15, further comprising the step of receiving, using said computer, a request to notify a recipient of the availability of said audible composition for playback.

18. A method as recited in claim 15, further comprising the step of transmitting using said computer an e-mail notification to said recipient that said composition is available for playback.

19. A method as recited in claim 18, wherein said computer comprises a web server, wherein said composition is stored on said web server in a plurality of data files, and wherein said e-mail notification contains a code for assembling said audio components into said composition.

20. A method as recited in claim 19, further comprising the step of said recipient transmitting said code to said web server.

21. A method as recited in claim 15, wherein said computer comprises a web server connected to a client computer through a communications link.

22. A method as recited in claim 15, further comprising the step of using said selection of audio components to compose audio content for playback using web browser software.

23. A method as recited in claim 22, further comprising the step of said recipient downloading, using a client computer, said audio components and playing said audio components using said web browser software.

24. A method as recited in claim 15, further comprising the step of receiving, using said computer, a sequence for said selected audio components to

be assembled into said composition.

25. A method as recited in claim 24, wherein said composition comprises contiguous audio corresponding to said selected audio components and said sequence.

26. A method as recited in claim 15, wherein said composition comprises contiguous audio generated from said audio components.

27. A method as recited in claim 15, wherein said composition comprises a dynamic audio sequence generated from said audio components.

28. A method as recited in claim 15, wherein said audio components are mixable as a time-based sequence.

29. A method for online composition and playback of audio content, comprising the steps of:

(a) providing, using a host computer, a listing of selectable audio components for inclusion in an audible composition;

(b) receiving, on a client computer connected to said host computer through a communications link, a selection of audio components for inclusion in an audible composition;

(c) generating, using said host computer, a set of data parameters corresponding to audible composition;

(d) receiving, from said client computer, a request to notify a recipient that said audible composition is available for playback; and

(e) notifying said recipient using e-mail sent from said host computer that said composition is available for playback.

30. A method as recited in claim 29, wherein said e-mail includes a code which controls assembly of said audio components into an audio sequence corresponding to said composition.

31. A method as recited in claim 30, further comprising the step of said recipient, using a client computer, transmitting to said host computer a request to playback said composition based on said downloaded parameters.

32. A method as recited in claim 31, further comprising the step of said recipient playing back said composition using web browser software on a client computer.

33. A method as recited in claim 32, wherein said composition is stored on said host computer in a plurality of data files, wherein said e-mail includes a code which controls assembly of said audio components into said composition, and further comprising the step of said recipient sending said code to said host computer.

34. A method as recited in claim 33, wherein said composition comprises contiguous audio corresponding to said selected audio components and said sequence.

35. A method as recited in claim 33, wherein said composition comprises contiguous audio generated from said audio components.

36. A method as recited in claim 33, wherein said composition comprises a dynamic audio sequence generated from said audio components.

37. A system for online composition and playback of audio content, comprising:

(a) a programmable data processor; and

(b) programming associated with said programmable data processor for carrying out the operations of

(i) providing a listing of selectable audio components for inclusion in an audible composition,

(ii) receiving a selection of audio components for inclusion in an audible composition, and

(iii) receiving a request to generate an audible composition corresponding to selected audio components.

38. A system as recited in claim 37, further comprising programming for generating data parameters for assembly of said audible composition in response to said request.

39. A system as recited in claim 37, further comprising programming for receiving a request to notify a recipient of the availability of said audible composition for playback.

40. A method as recited in claim 37, further comprising programming for transmitting an e-mail notification to said recipient that said composition is available for playback.

41. A system as recited in claim 40, wherein said e-mail notification includes a code which controls assembly of said audio components into an audio sequence corresponding to said composition.

42. A system as recited in claim 41, further comprising programming for receiving said code from said recipient and transmitting to said recipient selected audio components corresponding to said code for assembly into said composition.

43. A system as recited in claim 37, further comprising programming for receiving a sequence for said selected audio components to be assembled into said composition.

44. A system as recited in claim 43, wherein said composition comprises contiguous audio corresponding to said selected audio components and said sequence.

45. A system as recited in claim 37, wherein said composition comprises contiguous audio generated from said audio components.

46. A system as recited in claim 37, wherein said composition comprises a dynamic audio sequence generated from said audio components.

47. A system for online composition and playback of audio content, comprising:

(a) a web application server computer; and

(b) programming associated with said web application server for carrying out the operations of

(i) providing to a client computer a listing of selectable audio components for inclusion in an audible composition,

(ii) receiving from said client computer a selection of audio components for inclusion in an audible composition, and

(iii) receiving from said client computer a request to generate an audible composition corresponding to selected audio components.

48. A system as recited in claim 47, further comprising programming for receiving from said client computer a request to notify a recipient of the availability of said audible composition for playback.

49. A system as recited in claim 47, further comprising programming for transmitting from said web application server an e-mail notification to said recipient that said composition is available for playback.

50. A system as recited in claim 49, wherein said e-mail notification includes a code which controls assembly of said audio components into an audio sequence corresponding to said composition.

51. A system as recited in claim 47, wherein said selection of audio components is used to compose audio content for playback using web browser software.

52. A system as recited in claim 51, wherein said audio components are stored on said web application server, and further comprising programming for receiving said code from said recipient.

53. A system as recited in claim 52, further comprising programming for transmitting to said recipient audio components corresponding to said code.

54. A system as recited in claim 47, further comprising programming for receiving a sequence for said selected audio components to be assembled into said composition.

55. A system as recited in claim 54, wherein said composition comprises contiguous audio corresponding to said selected audio components and said sequence.

56. A system as recited in claim 47, wherein said composition comprises contiguous audio generated from said audio components.

57. A system as recited in claim 47, wherein said composition comprises a dynamic audio sequence generated from said audio components.

58. A system for online composition and playback of audio content, comprising:

(a) a host computer;

(b) a client computer; and

(c) programming associated with said host computer for carrying out the operations of

(i) providing to said client computer a listing of selectable audio components for inclusion in an audible composition,

(ii) receiving from said client computer a selection of audio components for inclusion in an audible composition,

(iii) generating a set of data parameters corresponding to audible composition,

(iv) receiving, from said client computer, a request to notify a recipient that said audible composition is available for playback, and

(v) notifying said recipient that said composition is available for playback.

59. A system as recited in claim 58, further comprising programming associated with said host computer for sending a code to said recipient which control assembly of said audio components into an audio sequence corresponding to said composition.

60. A system as recited in claim 59, further comprising programming associated with said host computer for receiving said code from said recipient.

61. A system as recited in claim 60, further comprising programming associated with said host computer for transmitting to said recipient audio components for assembly into said composition in response to receiving said code.

62. A system as recited in claim 61, wherein said composition comprises contiguous audio corresponding to said selected audio components and said sequence.

63. A system as recited in claim 61, wherein said composition comprises contiguous audio generated from said audio components.

64. A system as recited in claim 61, wherein said composition comprises a dynamic audio sequence generated from said audio components.

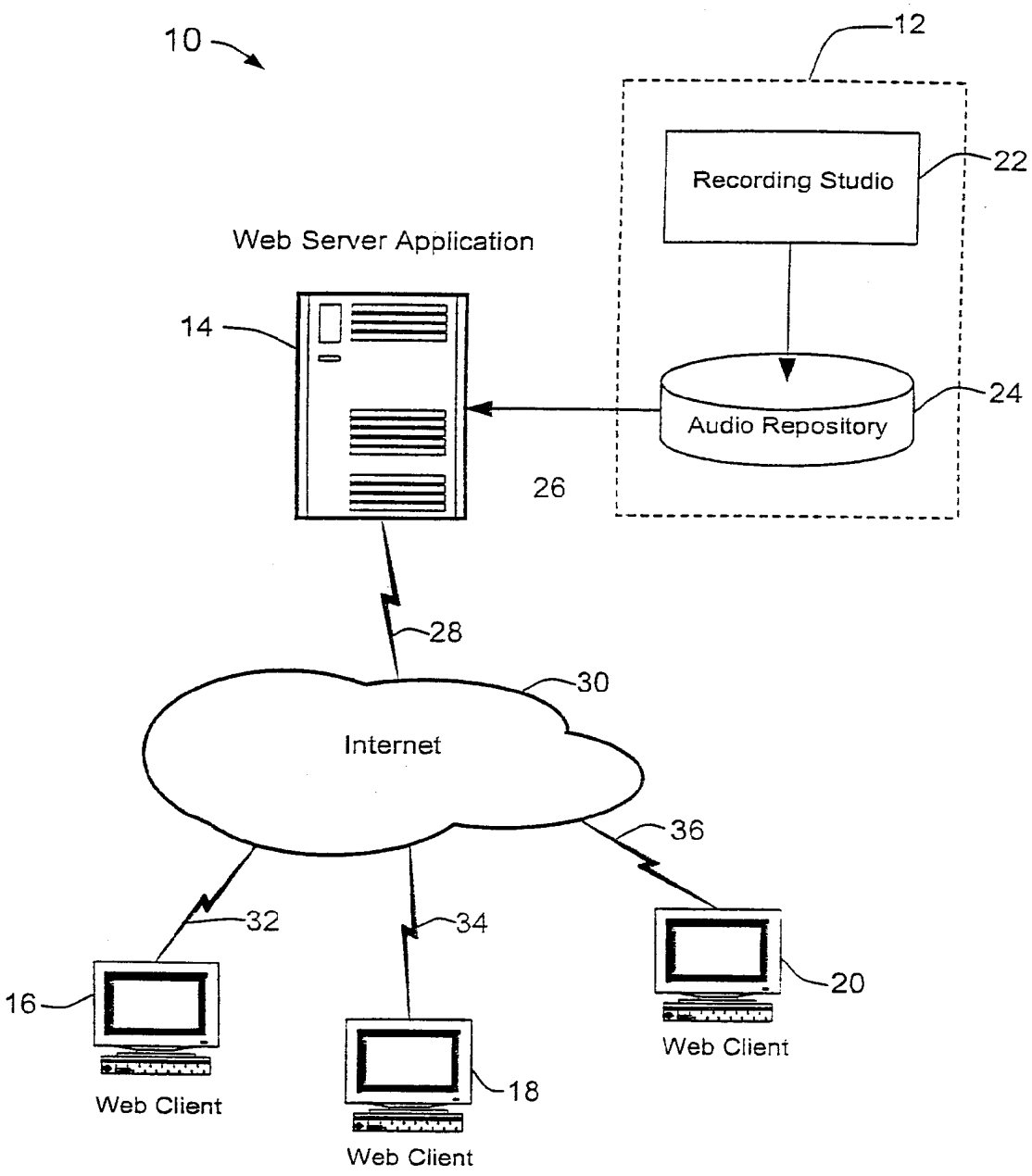
10
12
Recording Studio
22
Web Server Application
14
Audio Repository
24
26
28
30
Internet
36
32
34
16
20
Web Client
Web Client
18
Web Client

FIG. - 1

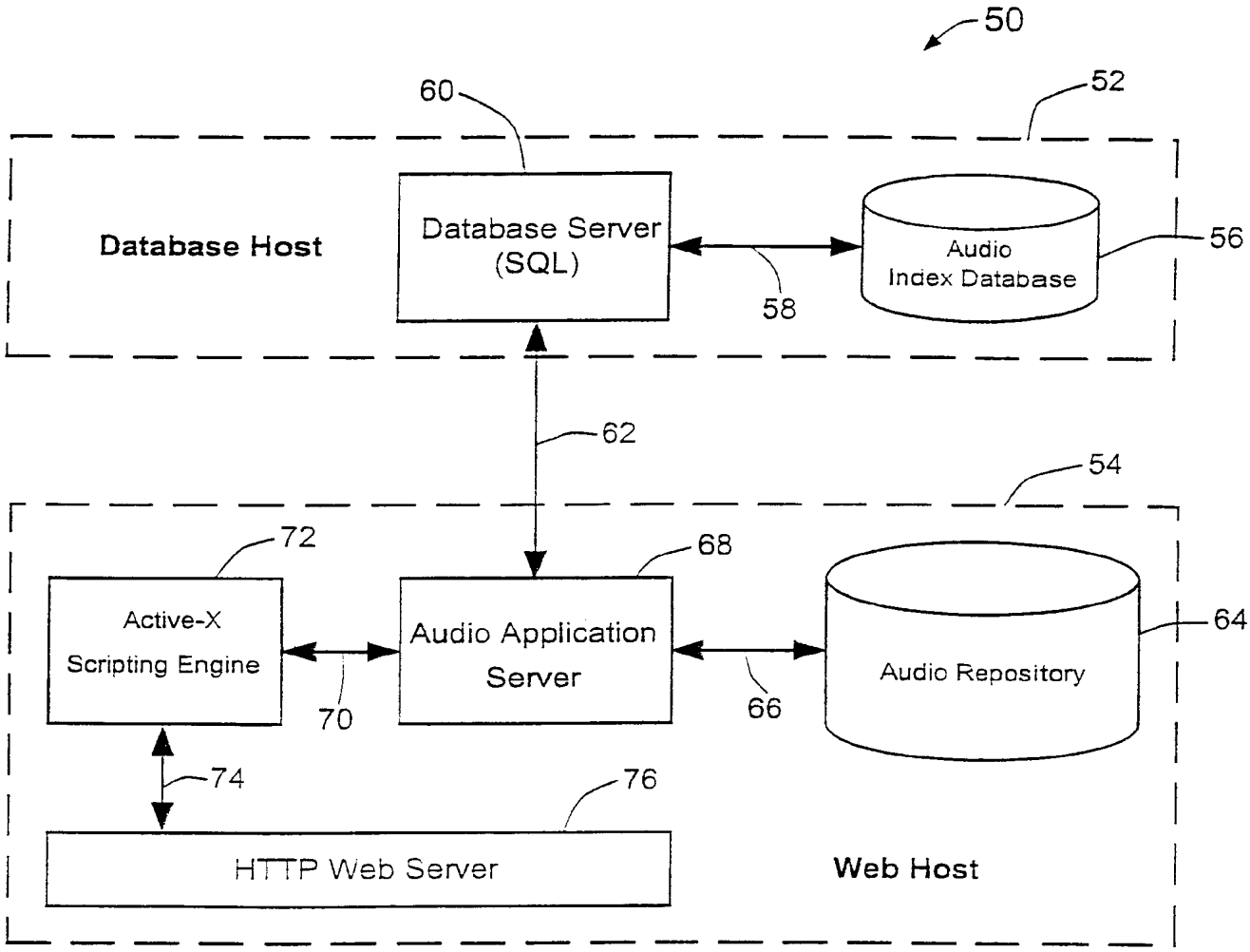
50
60
52
Database Host
Database Server
(SQL)
Audio
Index Database
56
58
62
54
72
68
Active-X
Scripting Engine
Audio Application
Server
Audio Repository
64
70
66
74
76
HTTP Web Server
Web Host

FIG. - 2

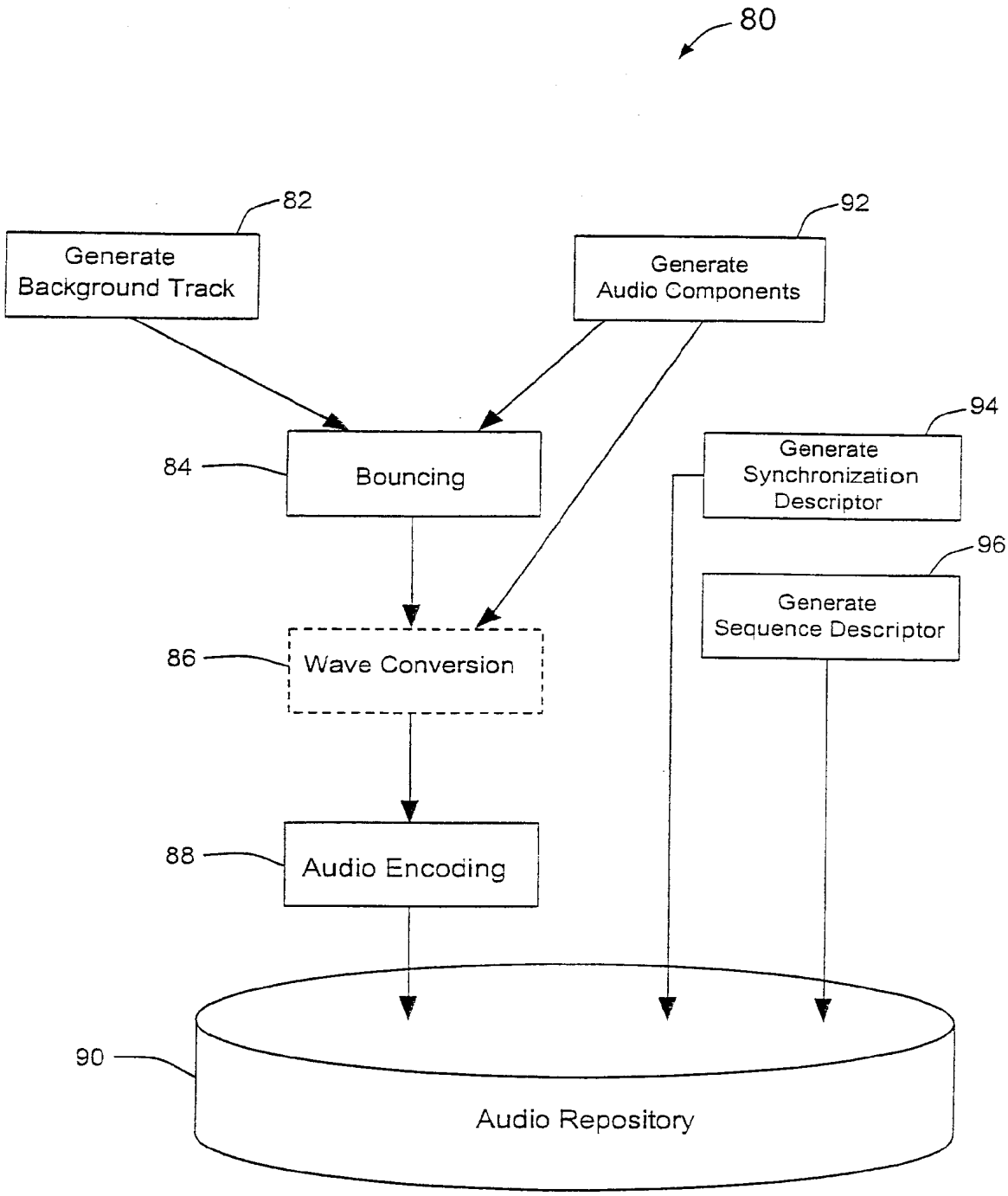
80
82
Generate Background Track
92
Generate Audio Components
84
Bouncing
94
Generate Synchronization Descriptor
96
Generate Sequence Descriptor
86
Wave Conversion
88
Audio Encoding
90
Audio Repository

FIG. - 3

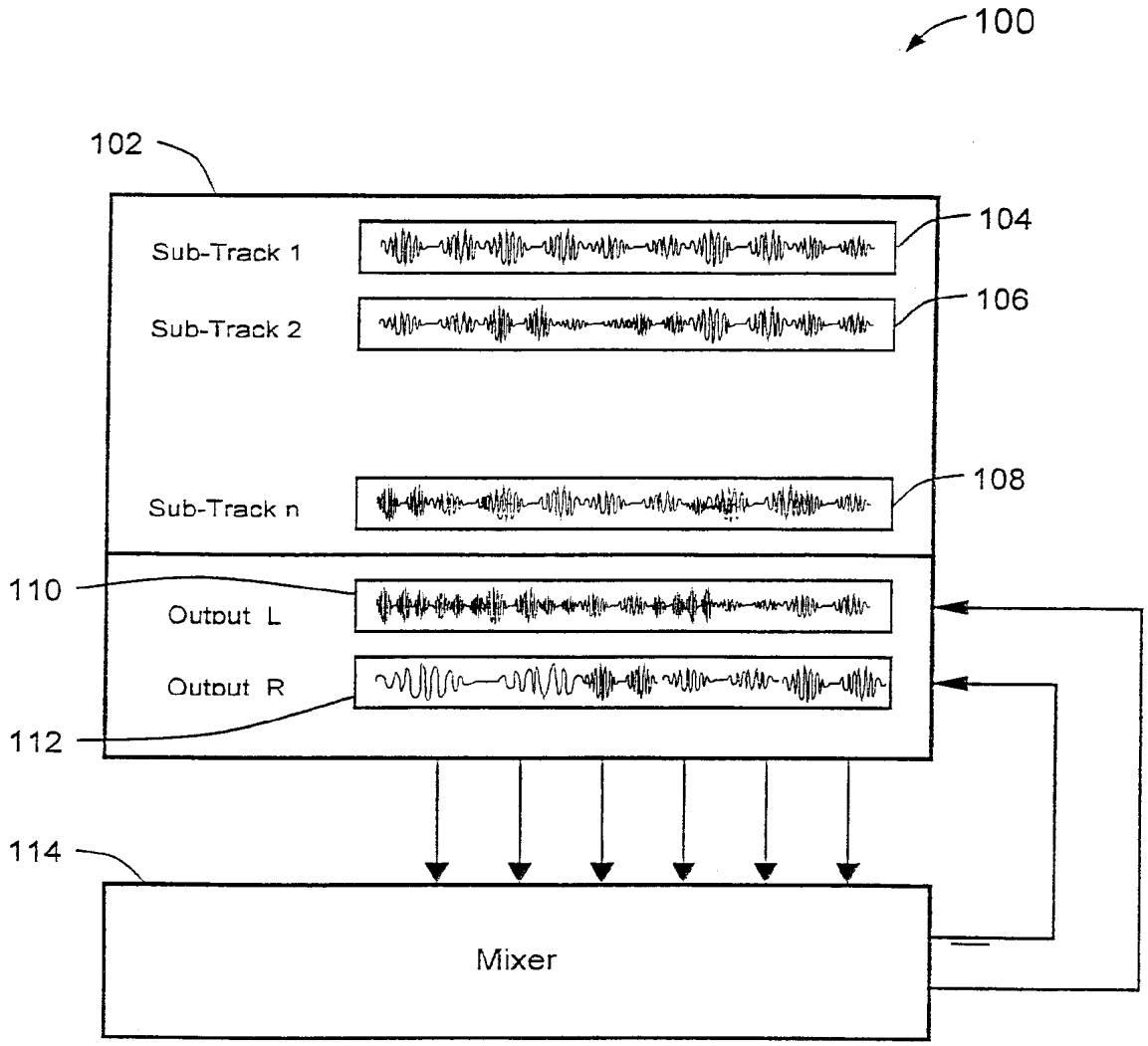
100
102
Sub-Track 1
104
Sub-Track 2
106
Sub-Track n
108
110
Output L
Output R
112
114
Mixer

FIG. - 4

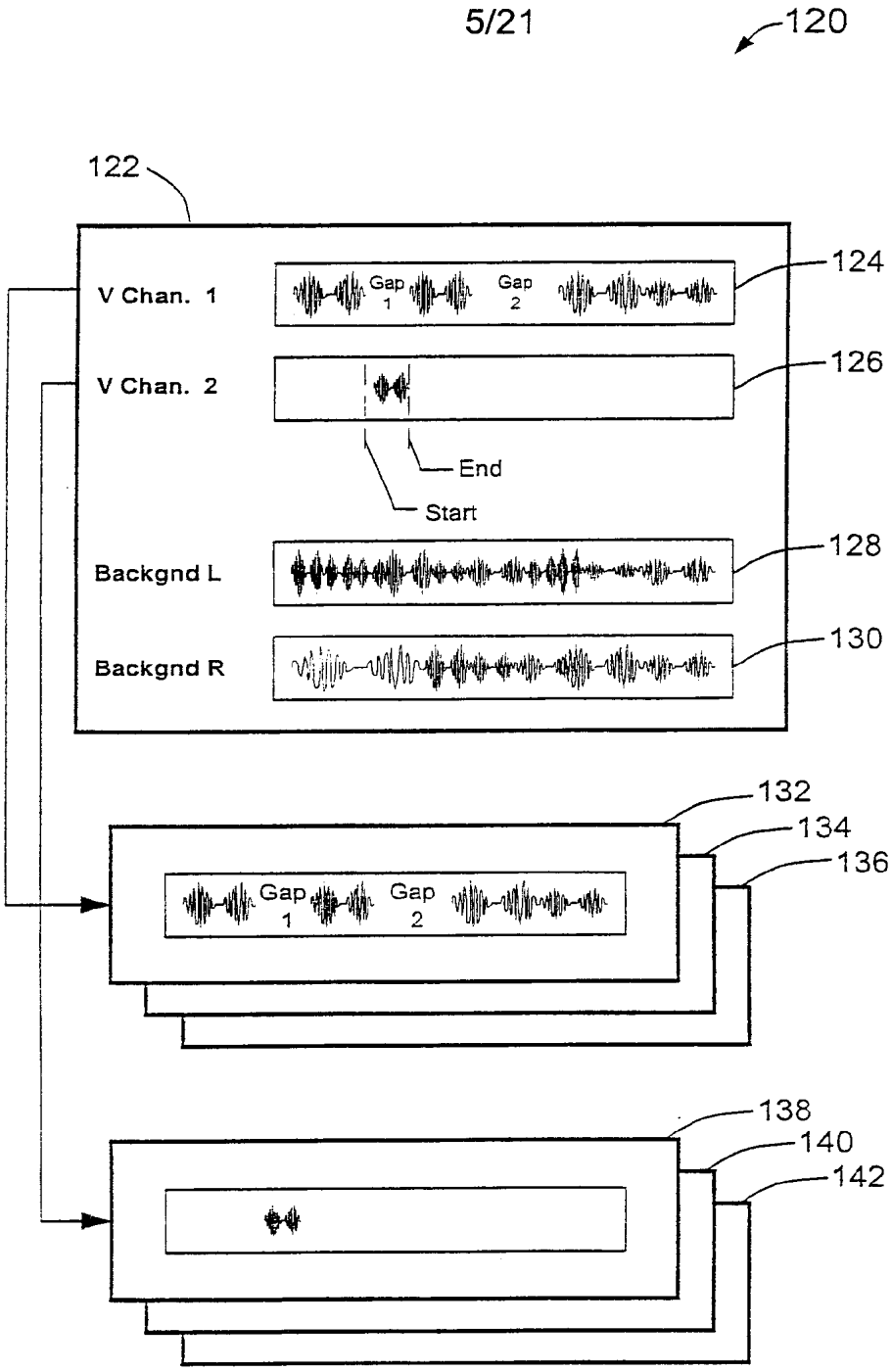
120
122
V Chan. 1
Gap 1
Gap 2
124
V Chan. 2
126
End
Start
Backgnd L
128
Backgnd R
130
132
134
136
Gap 1
Gap 2
138
140
142

FIG. - 5

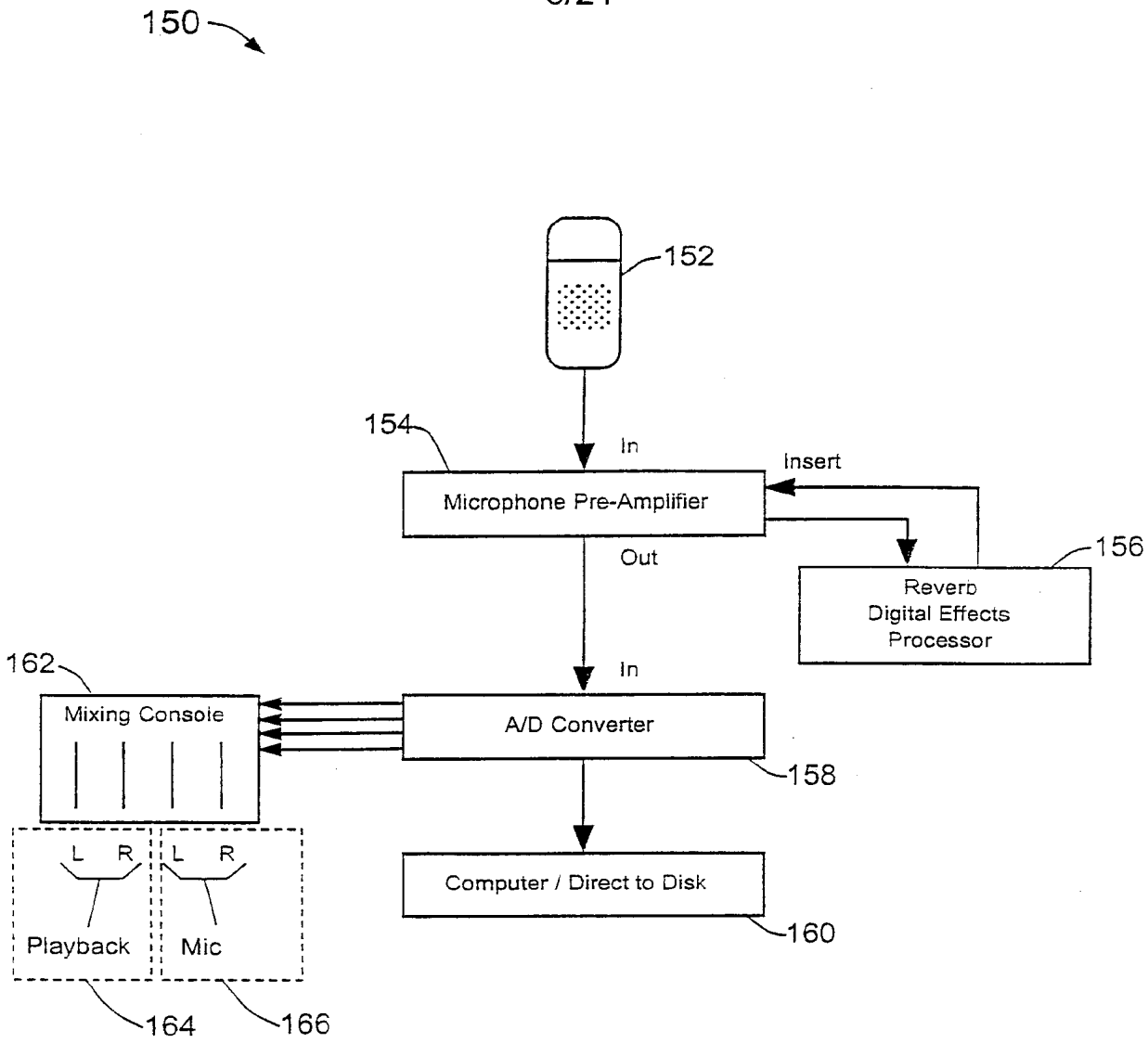
150
152
154
In
Insert
Microphone Pre-Amplifier
Out
156
Reverb
Digital Effects
Processor
In
162
Mixing Console
A/D Converter
158
L R
L R
Computer / Direct to Disk
160
Playback
Mic
164
166

FIG. - 6

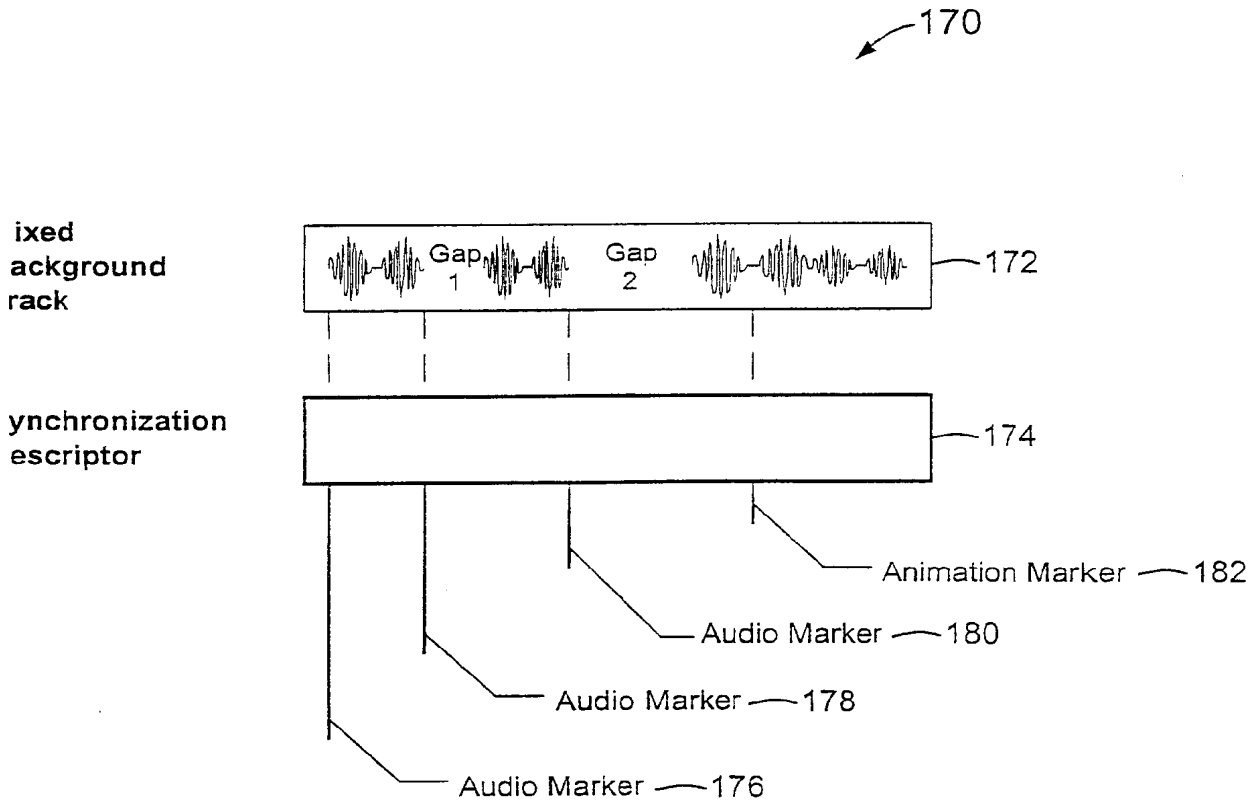
170
ixed
ackground
rack
Gap 1
Gap 2
172
ynchronization
escriptor
174
Animation Marker 182
Audio Marker 180
Audio Marker 178
Audio Marker 176

FIG. - 7

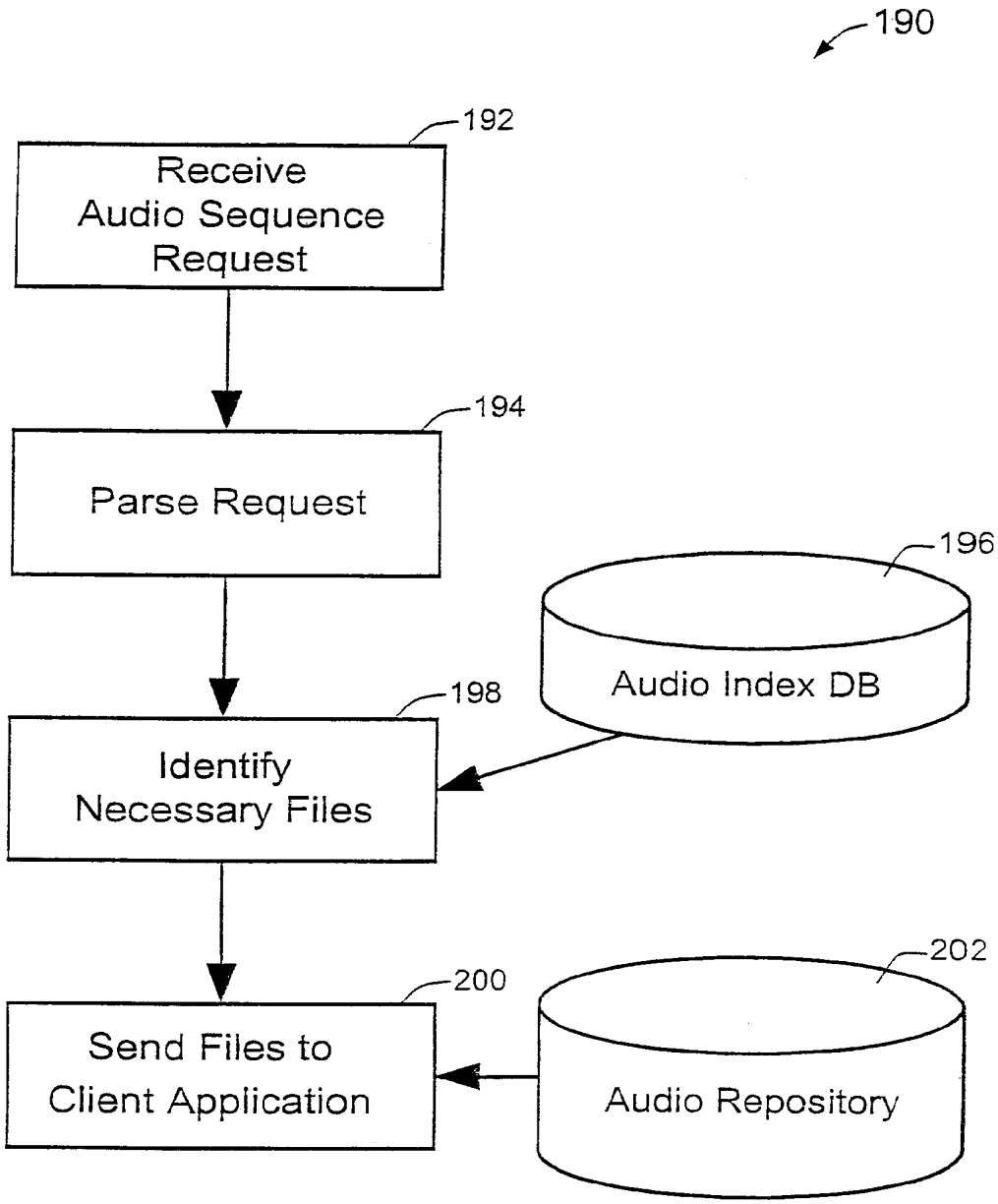
190
192
Receive Audio Sequence Request
194
Parse Request
196
Audio Index DB
198
Identify Necessary Files
200
Send Files to Client Application
202
Audio Repository

FIG. - 8

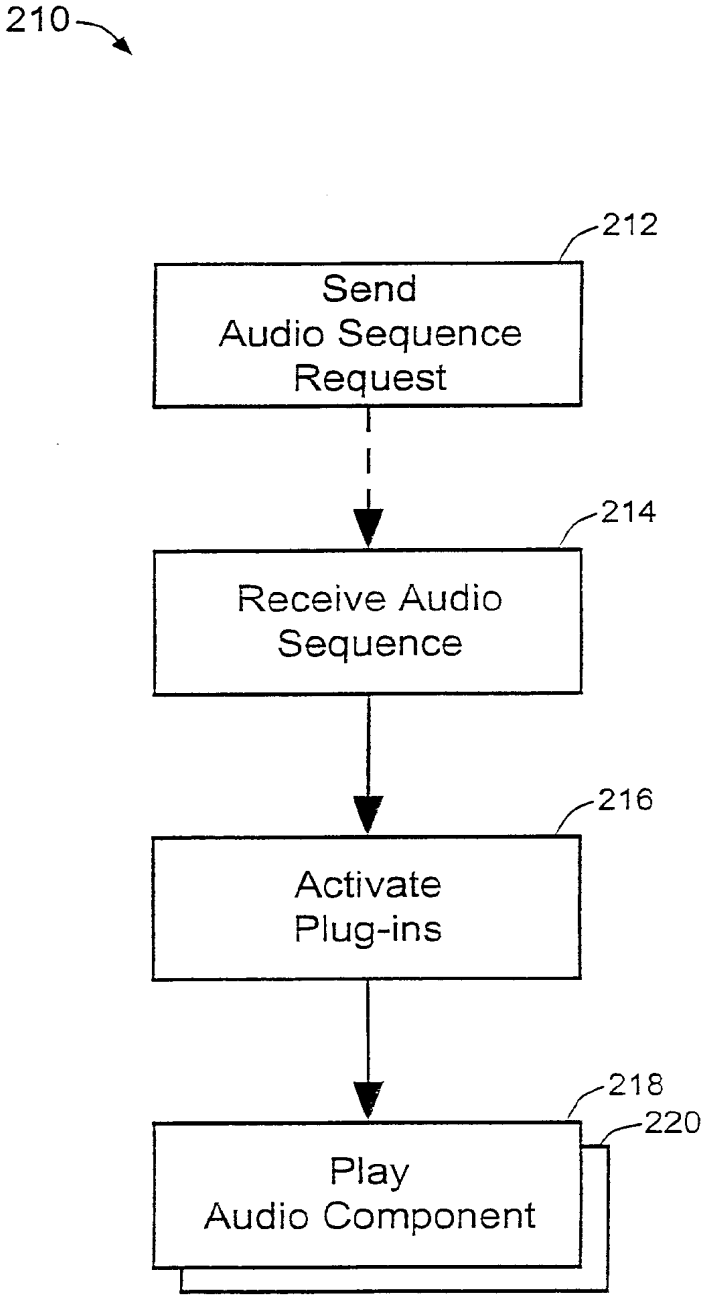
210
212
Send
Audio Sequence
Request
214
Receive Audio
Sequence
216
Activate
Plug-ins
218
220
Play
Audio Component

FIG. - 9

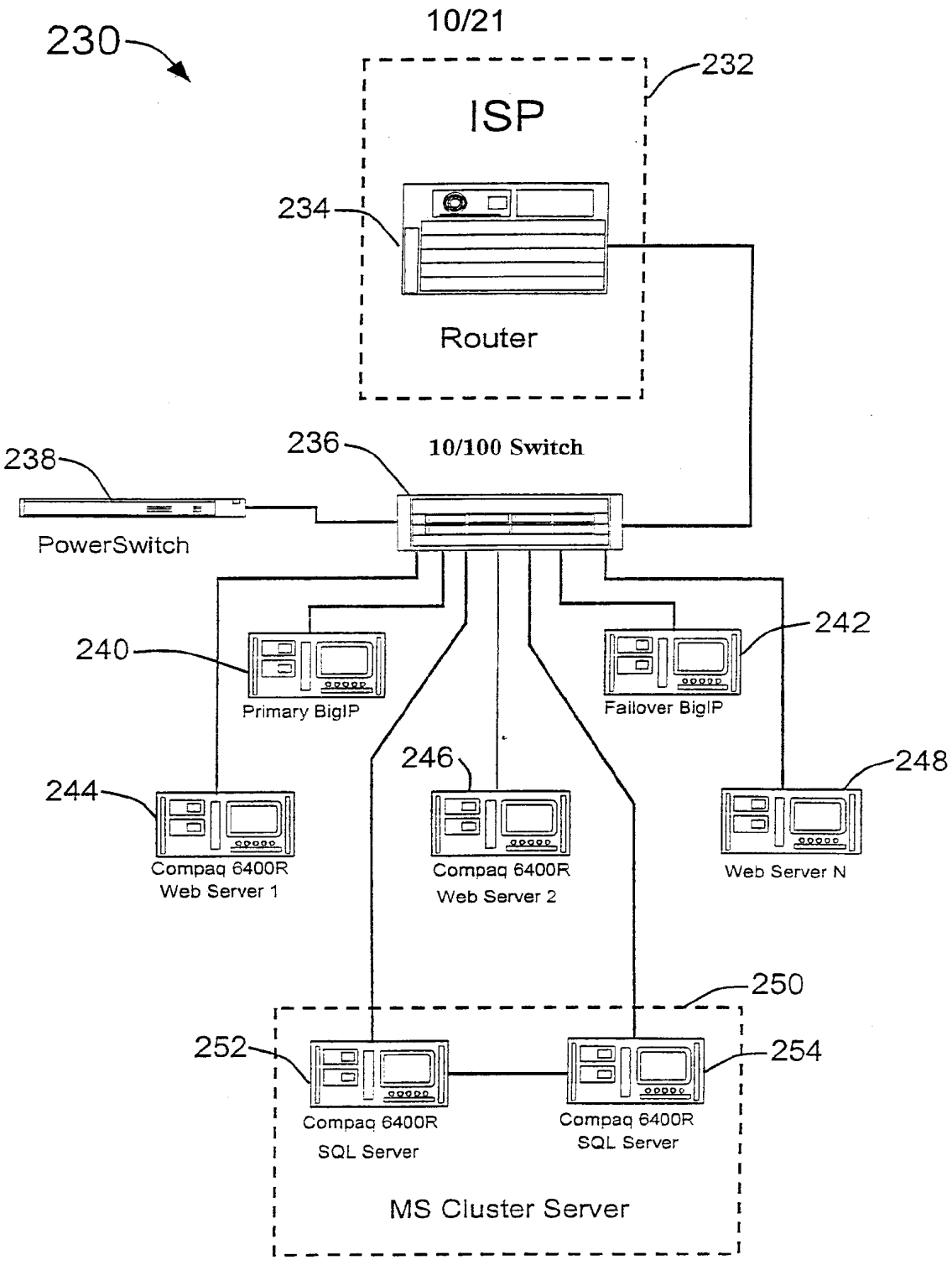
10/21
230
232
ISP
234
Router
236
10/100 Switch
238
PowerSwitch
240
Primary BigIP
242
Failover BigIP
244
Compaq 6400R
Web Server 1
246
Compaq 6400R
Web Server 2
248
Web Server N
250
252
Compaq 6400R
SQL Server
254
Compaq 6400R
SQL Server
MS Cluster Server

FIG. - 10

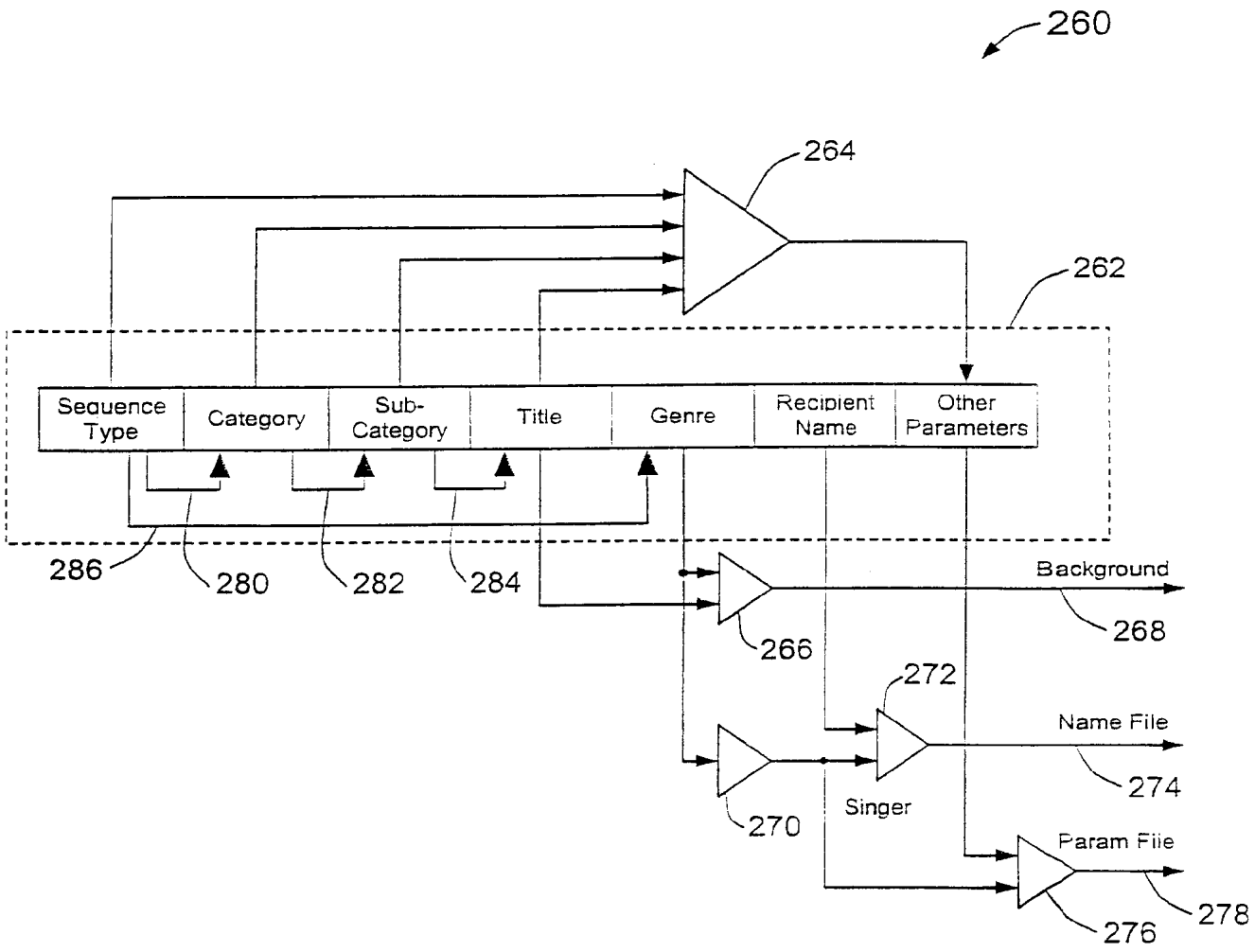
260
264
262
Sequence Type
Category
Sub-Category
Title
Genre
Recipient Name
Other Parameters
286
280
282
284
Background
266
268
272
Name File
274
Singer
270
Param File
278
276

FIG. 11

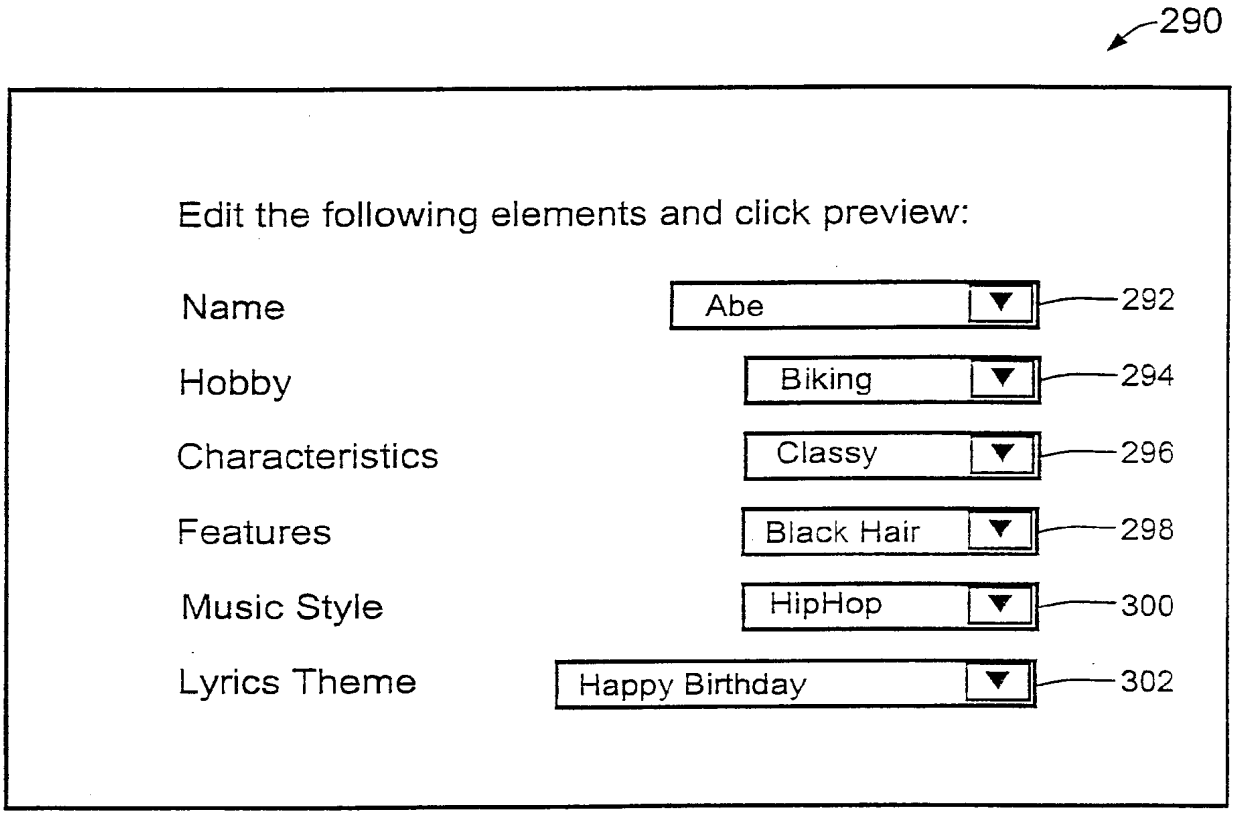
290
Edit the following elements and click preview:
Name
Abe
292
Hobby
Biking
294
Characteristics
Classy
296
Features
Black Hair
298
Music Style
HipHop
300
Lyrics Theme
Happy Birthday
302

FIG. 12

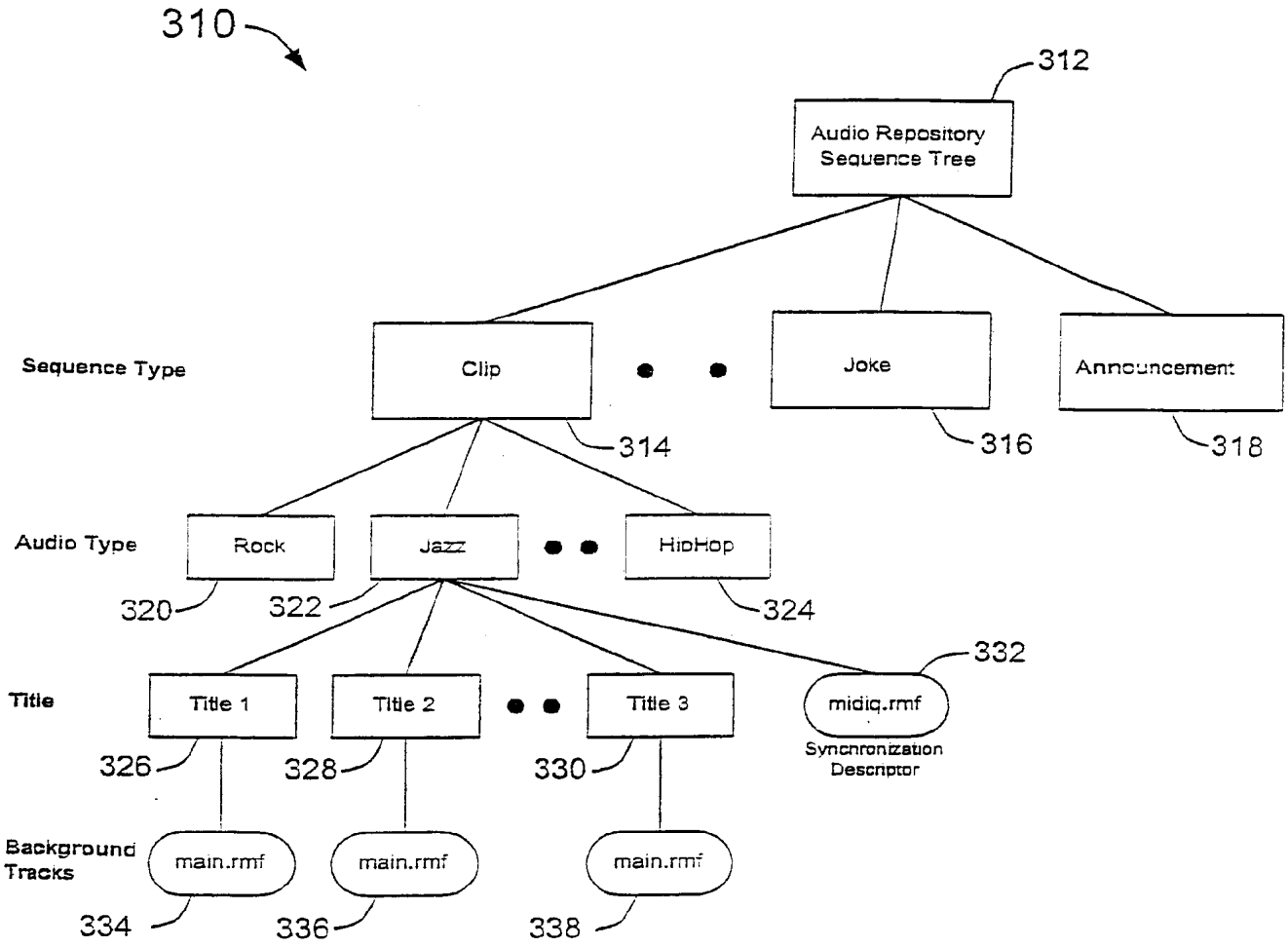
310
312
Audio Repository Sequence Tree
Sequence Type
Clip
Joke
Announcement
314
316
318
Audio Type
Rock
Jazz
HipHop
320
322
324
332
Title
Title 1
Title 2
Title 3
midiq.rmf
Synchronization Descriptor
326
328
330
Background Tracks
main.rmf
main.rmf
main.rmf
334
336
338

FIG. 13

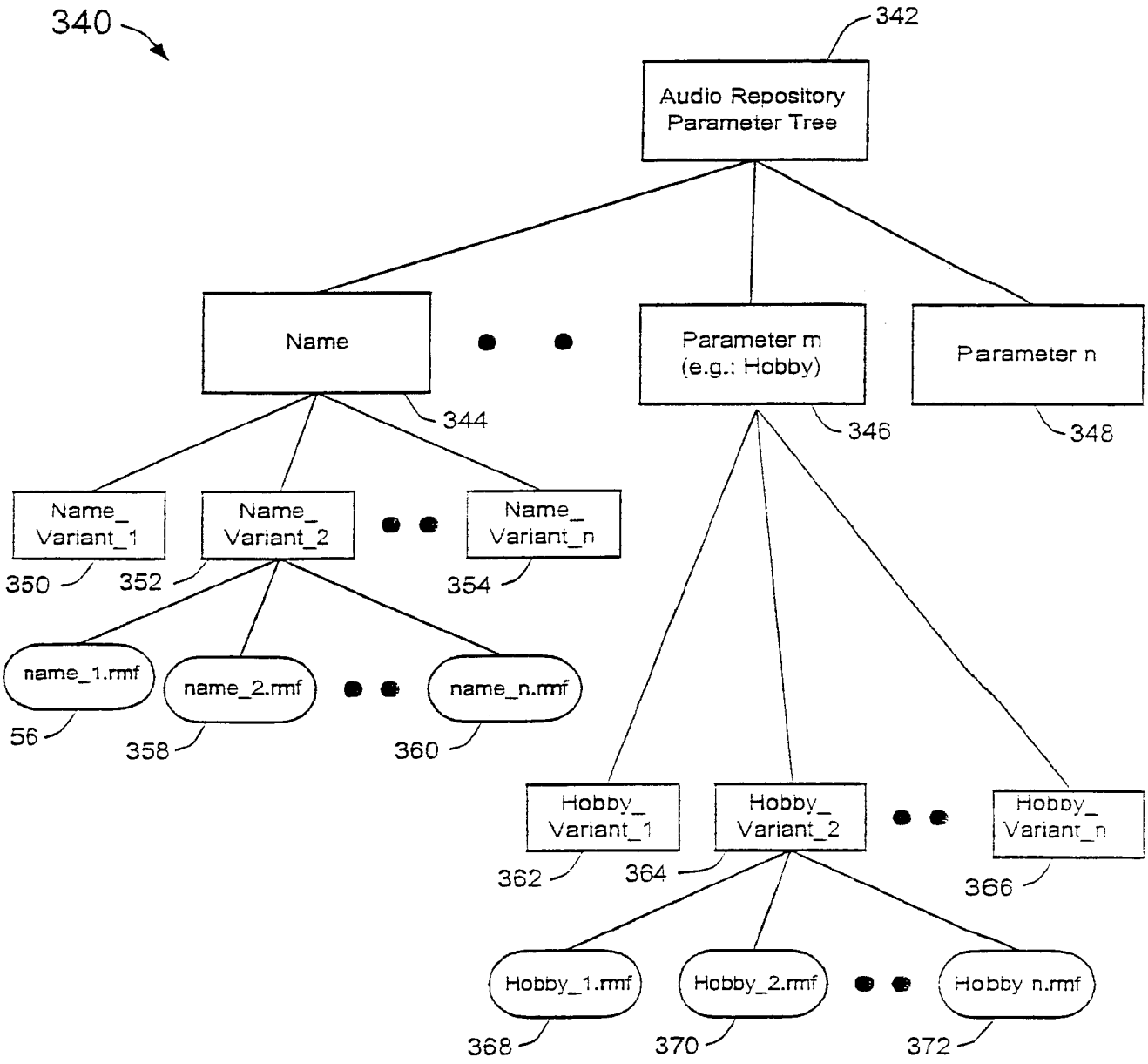
340
342
Audio Repository Parameter Tree
Name
344
Parameter m (e.g.: Hobby)
346
Parameter n
348
Name_ Variant_1
350
Name_ Variant_2
352
Name_ Variant_n
354
name_1.rmf
56
name_2.rmf
358
name_n.rmf
360
Hobby_ Variant_1
362
Hobby_ Variant_2
364
Hobby_ Variant_n
366
Hobby_1.rmf
368
Hobby_2.rmf
370
Hobby n.rmf
372

FIG. - 14

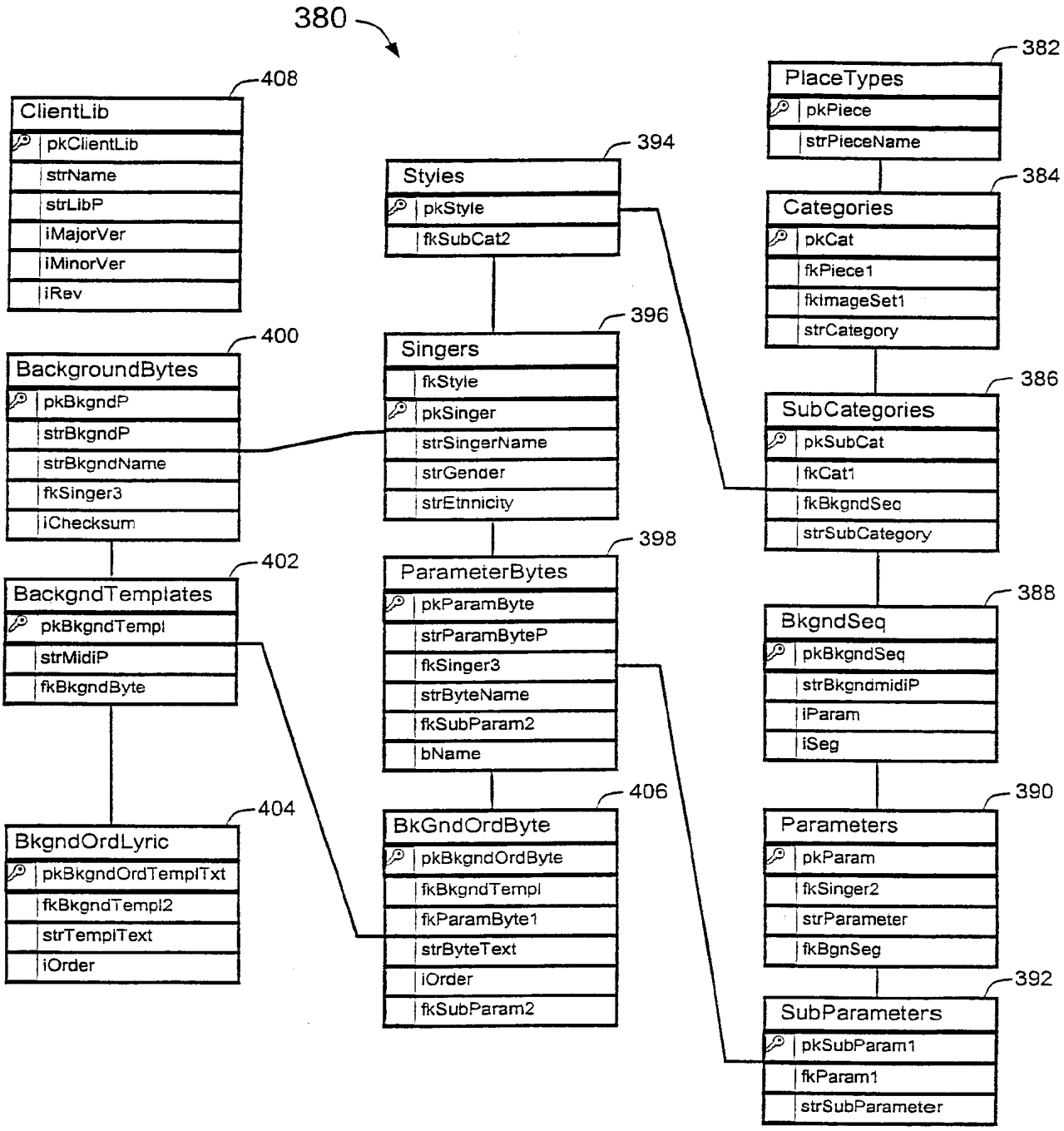
380
408
ClientLib
pkClientLib
strName
strLibP
iMajorVer
iMinorVer
iRev
400
BackgroundBytes
pkBkgndP
strBkgndP
strBkgndName
fkSinger3
iChecksum
402
BackgndTemplates
pkBkgndTempl
strMidiP
fkBkgndByte
404
BkgndOrdLyric
pkBkgndOrdTemplTxt
fkBkgndTempl2
strTemplText
iOrder
394
Styles
pkStyle
fkSubCat2
396
Singers
fkStyle
pkSinger
strSingerName
strGender
strEtnnicity
398
ParameterBytes
pkParamByte
strParamByteP
fkSinger3
strByteName
fkSubParam2
bName
406
BkGndOrdByte
pkBkgndOrdByte
fkBkgndTempl
fkParamByte1
strByteText
iOrder
fkSubParam2
382
PlaceTypes
pkPiece
strPieceName
384
Categories
pkCat
fkPiece1
fkimageSet1
strCategory
386
SubCategories
pkSubCat
fkCat1
fkBkgndSeq
strSubCategory
388
BkgndSeq
pkBkgndSeq
strBkgndmidiP
iParam
iSeg
390
Parameters
pkParam
fkSinger2
strParameter
fkBgnSeg
392
SubParameters
pkSubParam1
fkParam1
strSubParameter

FIG. - 15

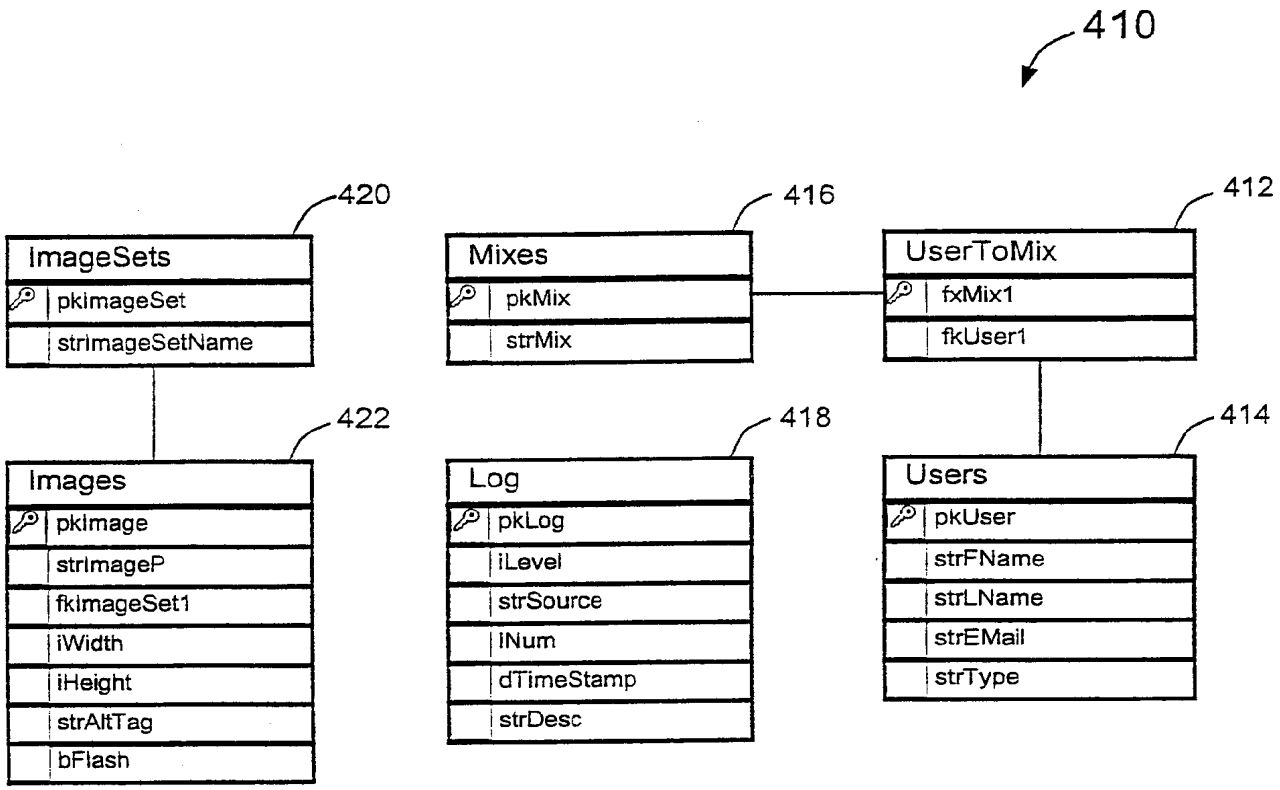
410
420
ImageSets
pkImageSet
strImageSetName
422
Images
pkImage
strImageP
fkImageSet1
iWidth
iHeight
strAltTag
bFlash
416
Mixes
pkMix
strMix
418
Log
pkLog
iLevel
strSource
iNum
dTimeStamp
strDesc
412
UserToMix
fxMix1
fkUser1
414
Users
pkUser
strFName
strLName
strEMail
strType

FIG. - 16

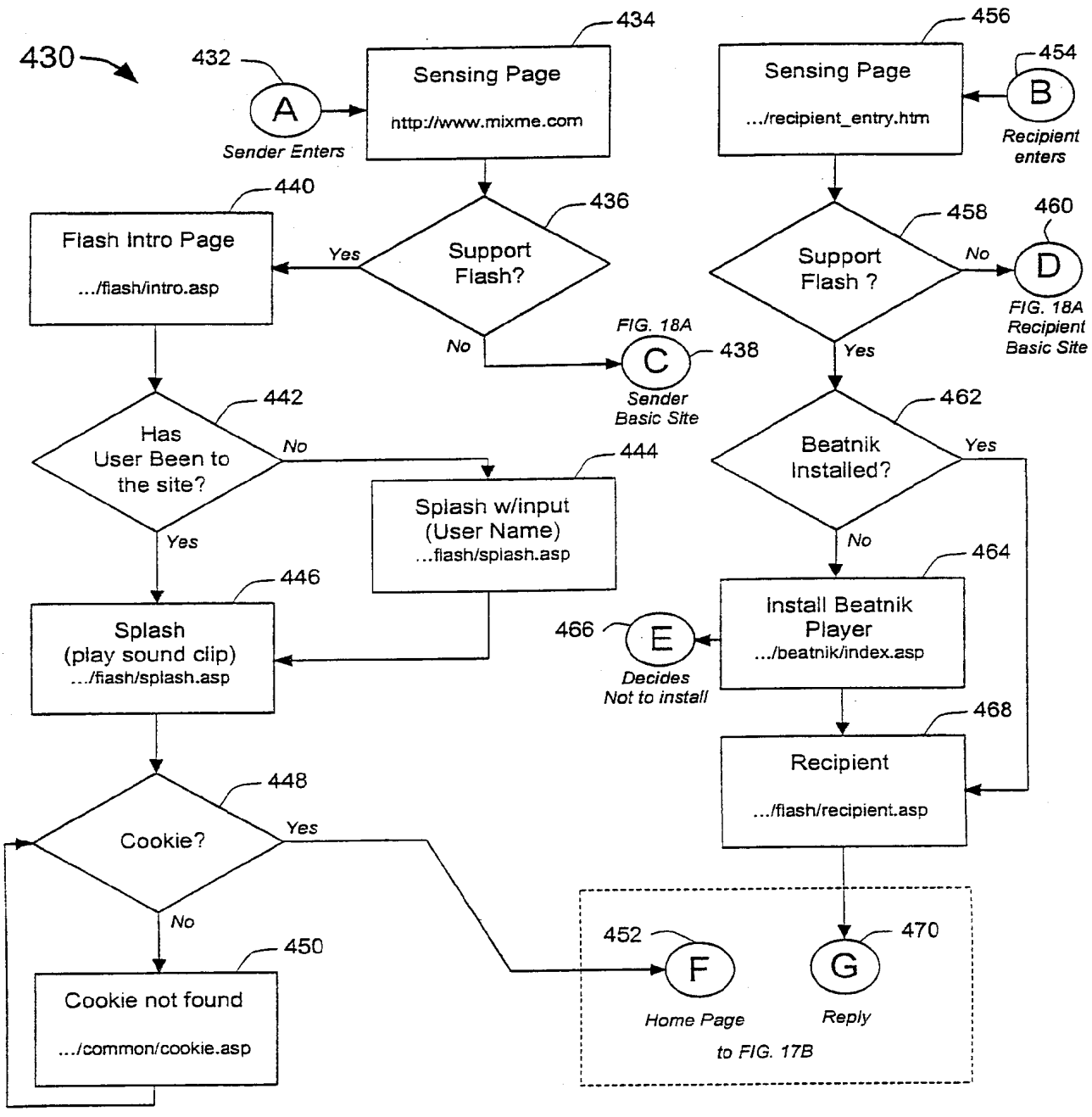
430
432
A
Sender Enters
434
Sensing Page
http://www.mixme.com
456
Sensing Page
.../recipient_entry.htm
454
B
Recipient enters
440
Flash Intro Page
.../flash/intro.asp
436
Support Flash?
Yes
No
458
Support Flash ?
No
460
D
FIG. 18A Recipient Basic Site
FIG. 18A
C
438
Sender Basic Site
Yes
442
Has User Been to the site?
No
Yes
444
Splash w/input (User Name)
...flash/splash.asp
462
Beatnik Installed?
Yes
No
446
Splash
(play sound clip)
.../flash/splash.asp
464
Install Beatnik Player
.../beatnik/index.asp
466
E
Decides Not to install
468
Recipient
.../flash/recipient.asp
448
Cookie?
Yes
No
450
Cookie not found
.../common/cookie.asp
452
F
Home Page
470
G
Reply
to FIG. 17B

FIG. – 17A

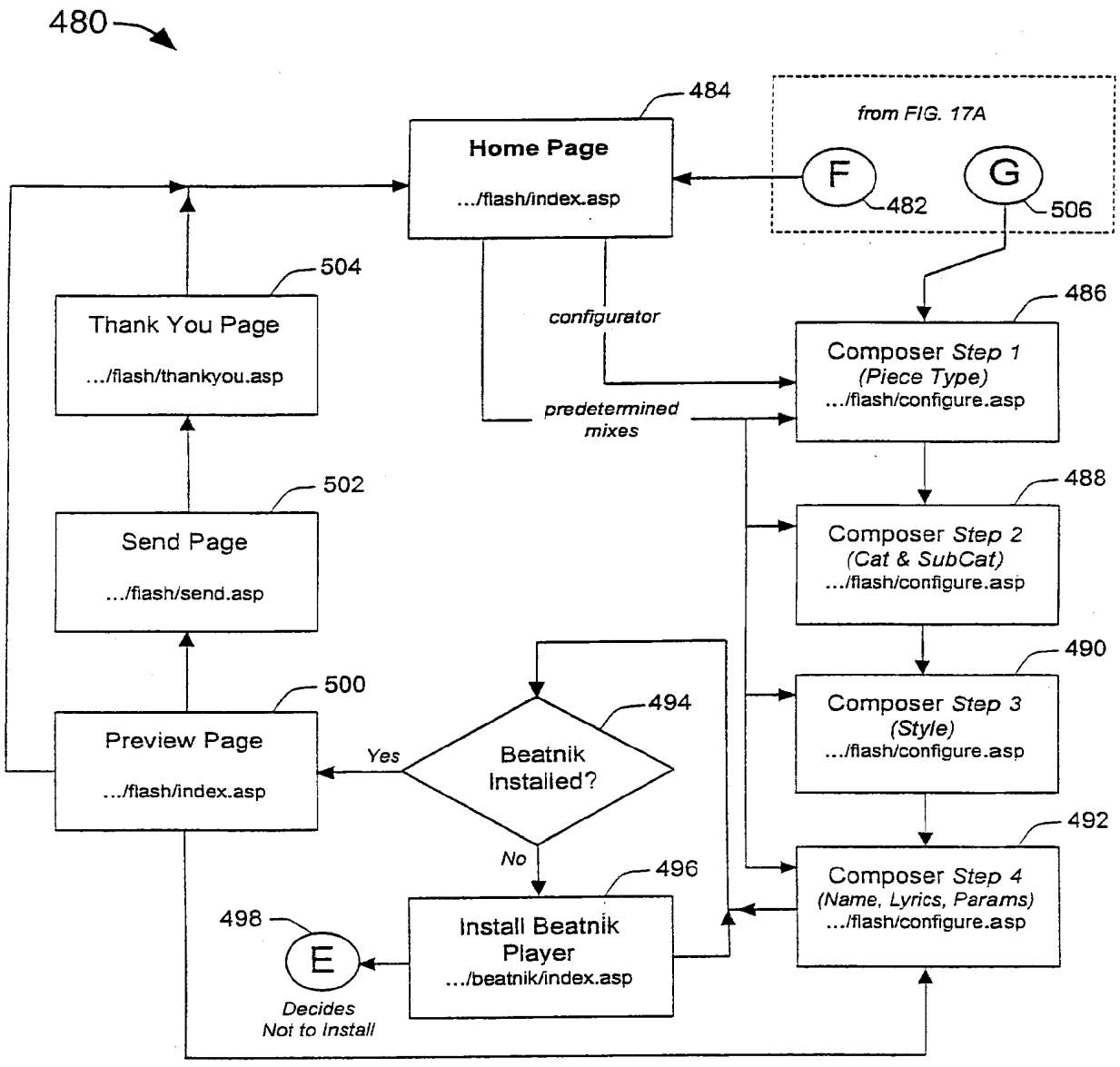
480
484
Home Page
.../flash/index.asp
from FIG. 17A
F
482
G
506
504
Thank You Page
.../flash/thankyou.asp
configurator
predetermined mixes
486
Composer Step 1
(Piece Type)
.../flash/configure.asp
502
Send Page
.../flash/send.asp
488
Composer Step 2
(Cat & SubCat)
.../flash/configure.asp
490
Composer Step 3
(Style)
.../flash/configure.asp
500
Preview Page
.../flash/index.asp
Yes
494
Beatnik Installed?
No
496
Composer Step 4
(Name, Lyrics, Params)
.../flash/configure.asp
492
498
E
Decides Not to Install
Install Beatnik Player
.../beatnik/index.asp

FIG. - 17B

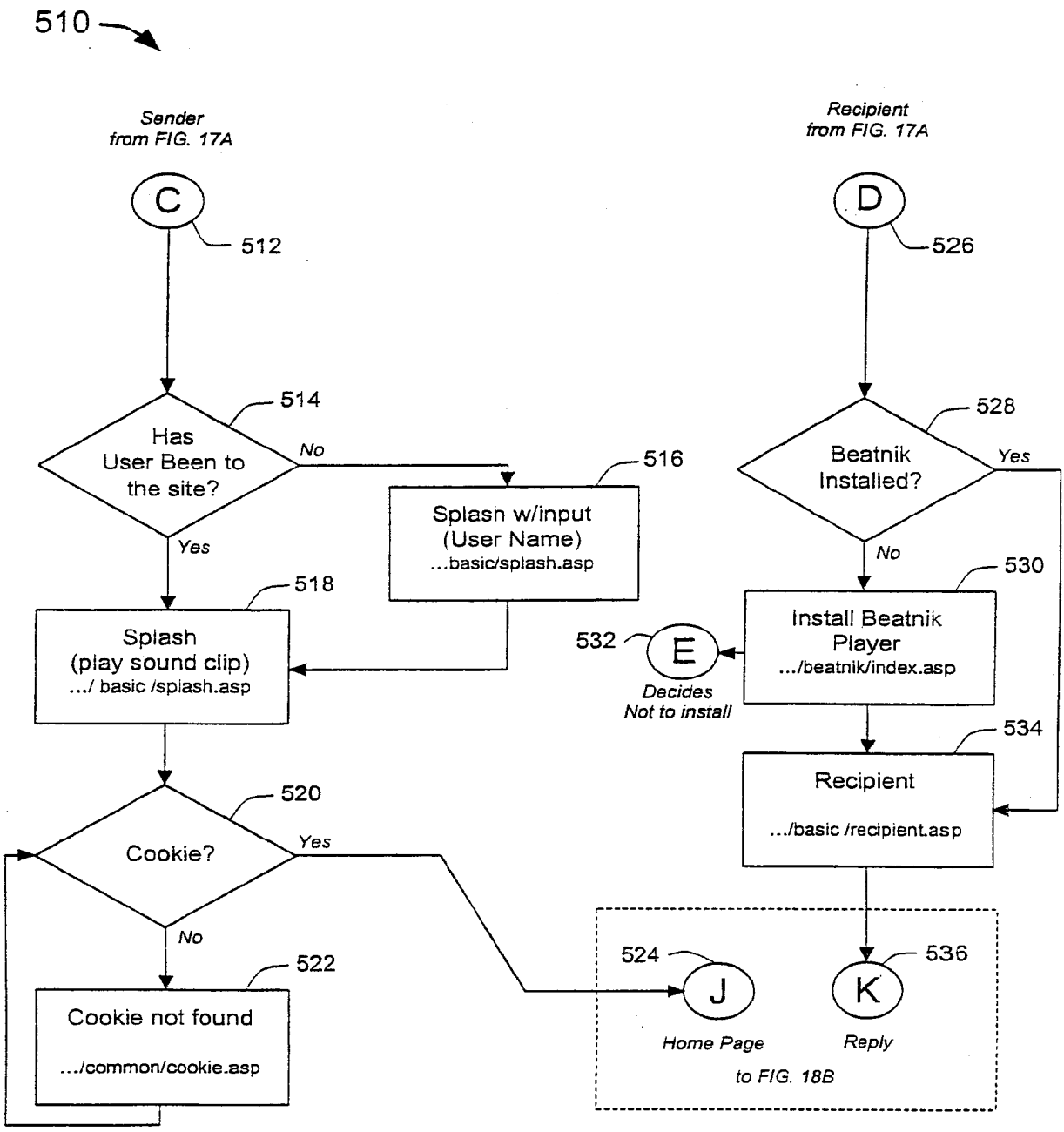
510
Sender
from FIG. 17A
Recipient
from FIG. 17A
C
512
D
526
Has
User Been to
the site?
514
No
Yes
516
Splasn w/input
(User Name)
...basic/splash.asp
Beatnik
Installed?
528
Yes
No
530
Install Beatnik
Player
.../beatnik/index.asp
532
E
Decides
Not to install
518
Splash
(play sound clip)
.../ basic /splash.asp
534
Recipient
.../basic /recipient.asp
Cookie?
520
Yes
No
522
Cookie not found
.../common/cookie.asp
524
J
Home Page
536
K
Reply
to FIG. 18B

FIG. – 18A

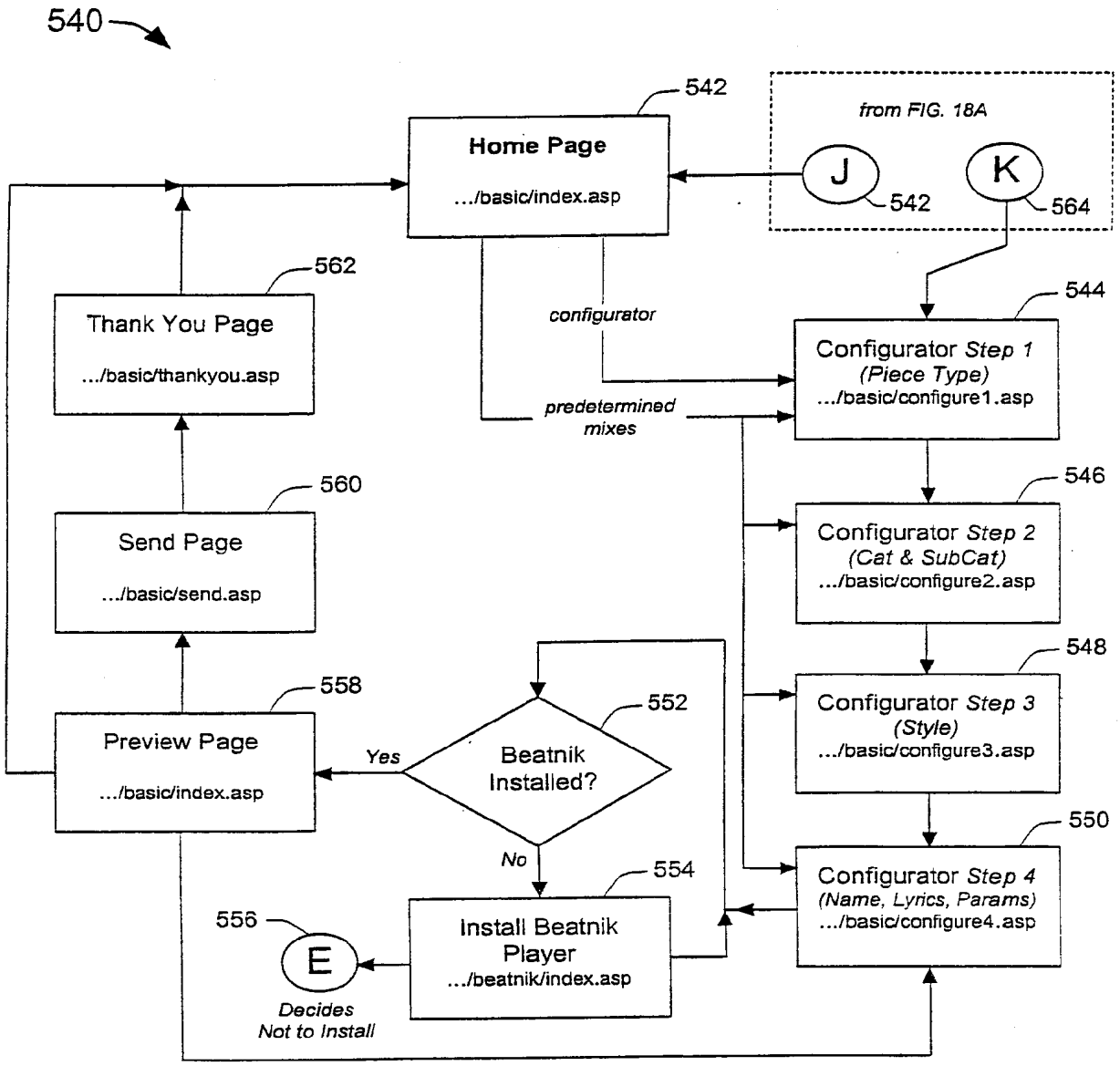
540
542
Home Page
.../basic/index.asp
from FIG. 18A
J
542
K
564
562
Thank You Page
.../basic/thankyou.asp
configurator
predetermined mixes
544
Configurator Step 1
(Piece Type)
.../basic/configure1.asp
560
Send Page
.../basic/send.asp
546
Configurator Step 2
(Cat & SubCat)
.../basic/configure2.asp
548
Configurator Step 3
(Style)
.../basic/configure3.asp
558
Preview Page
.../basic/index.asp
Yes
552
Beatnik Installed?
No
554
Install Beatnik Player
.../beatnik/index.asp
556
E
Decides Not to Install
550
Configurator Step 4
(Name, Lyrics, Params)
.../basic/configure4.asp

FIG. – 18B

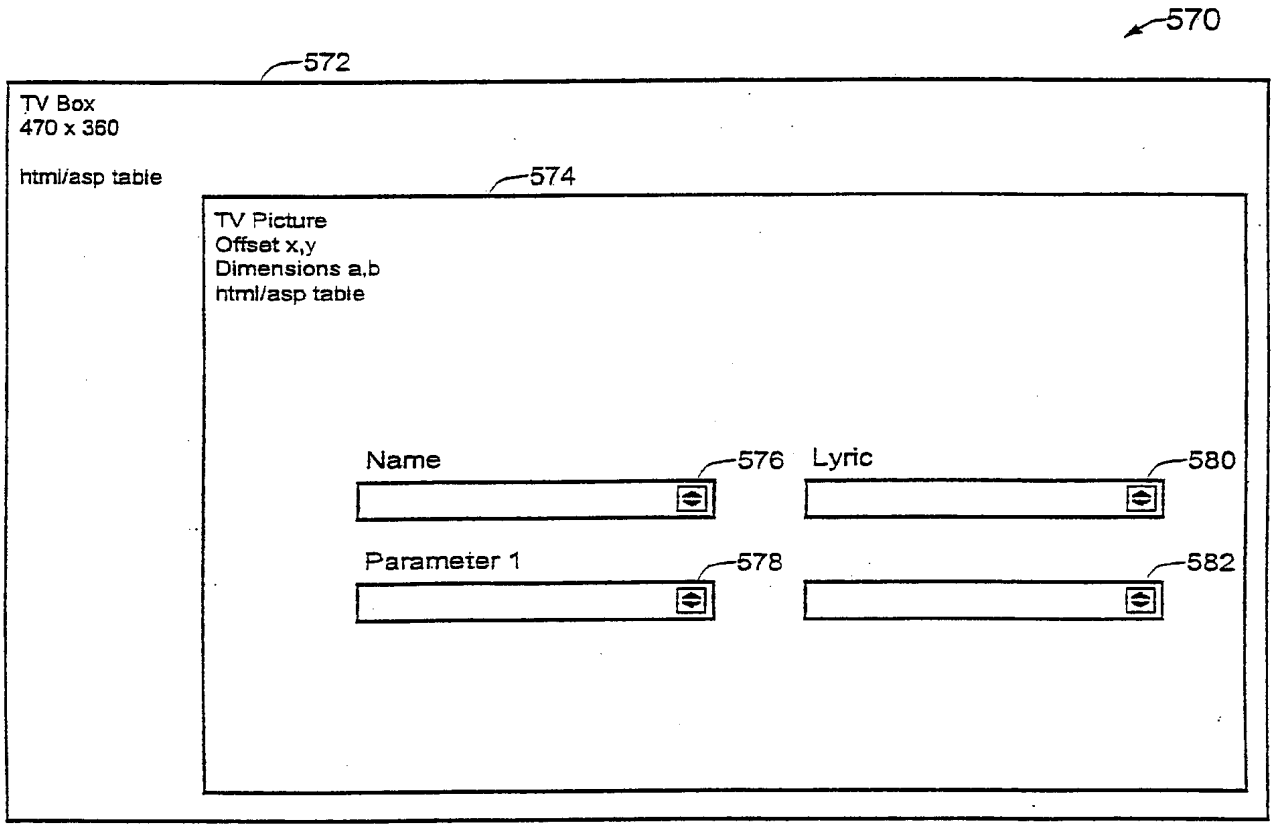
570
572
TV Box
470 x 360
html/asp table
574
TV Picture
Offset x,y
Dimensions a,b
html/asp table
Name
576
Lyric
580
Parameter 1
578
582

FIG. – 19

INTERNATIONAL SEARCH REPORT

Inte. onal Application No
PCT/US 00/21019

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G10H1/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G10H

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 886 274 A (JUNGLEIB STANLEY) 23 March 1999 (1999-03-23) column 1, line 53 -column 2, line 31 column 2, line 65 -column 3, line 14 column 8, line 12 - line 34; claims 1,11,36,38; figure 1	1,2,6, 15,16, 21,37, 38,47
A	US 5 590 282 A (CLYNES MANFRED) 31 December 1996 (1996-12-31) column 3, line 66 -column 2, line 21 column 4, line 46 -column 6, line 6; figure 1	1-64

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 November 2000	22/11/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Pulluard, R

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Inter onal Application No
PCT/US 00/21019

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 773 741 A (MILLS BRENT R ET AL) 30 June 1998 (1998-06-30) column 3, line 66 -column 2, line 20 column 5, line 51 -column 6, line 62 column 7, line 12 - line 26 column 7, line 40 - line 56; figure 1	1-64

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

Inter ›nal Application No
PCT/US 00/21019

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5886274 A	23-03-1999	JP 11073182 A	16-03-1999
US 5590282 A	31-12-1996	NONE	
US 5773741 A	30-06-1998	NONE	